Study on Numerical Laplace Transforms and Their Applications to Analysis of Transmission Lines(Dissertation_全文)

AUTHOR(S):

Yonemoto, Akihiro

Study on Numerical Laplace Transforms and Their Applications to Analysis of Transmission Lines

Akihiro Yonemoto

December 2003

Study on Numerical Laplace Transforms and Their Applications to Analysis of Transmission Lines

Akihiro Yonemoto

December 2003

Contents

Chapter 1

Introduction

1.1 Background

Transmission of information and energy is a major issue now more than ever. The semiconductor technology has been progressing at a steady exponential rate over the last few decades, and the progress has realized LSIs operating above some gigahertz. As clock speeds increase and process technology continues to advance, interconnect delays have surpassed gate delays within a chip. Hence development of high speed integrated circuits is in need of higher performance interconnects that are treated as distributed transmission lines. On the other hand, the demand for electricity has been growing rapidly along with the diffusion of information and electronic appliances. As electric power systems have become larger and more complex, the safety of the power systems has emerged as a critical issue, which is even more important if the power industry deregulation results in various kinds of power stations incorporated into existing power systems. In large-scale power networks, a fault can cause a disastrous power failure spreading over a large area. Once a fault occurs on power transmission lines, it is necessary to detect the fault and locate the fault point as soon as possible to prevent such a catastrophe.

Analysis of electric transients on transmission lines is fundamental in those areas. The interconnects in integrated circuits behave as transmission lines at high frequencies, and signal is degraded due to reflections, crosstalk, frequency-dependent losses, and so on. Then, it is essential to evaluate their effects on transient response, such as delay, rise time, ringing, etc., in order to maintain signal integrity. As for protection of power systems, a fault generates electric transients that propagate along power transmission lines and contain information in a certain manner about when and where the fault occurred, and what kind of a fault it was. Though the analysis of

fault transients requires us to treat the power transmission lines as distributed transmission lines instead of the simplified lumped circuits, it will lead to a more robust and accurate fault location method.

Laplace transforms provide a powerful tool to analyze transient behavior of time-invariant linear systems and are widely used in electrical engineering, control systems, chemistry engineering, mechanical engineering, and probabilistic theory. Just like a Fourier transform, a Laplace transform gives a representation of signals in the frequency domain, which representation is of great utility when solving differential equations and/or taking into account frequency-dependent effects. For example, transient waveforms on transmission lines are described by the telegraph equation and are significantly distorted by the skin effect and the dielectric loss, that are proportional to $\sqrt{f}$ and f, respectively, where f denotes the frequency. It is, however, difficult to invert Laplace transforms analytically especially when they contain the frequency-dependent terms, thereby numerical methods are required to compute Laplace inversions.

Numerical inversion of Laplace transforms has been studied extensively, and various kinds of methods have been proposed thus far [1, 2]. Some of them utilize Legendre functions [3], Laguerre functions [4], Gaussian quadrature [5], and so forth [6, 7]. After the fast Fourier transform algorithm was devised by J. W. Cooley and O. W. Tukey in 1965, Fourier series method using the FFT algorithm has been proposed in a number of papers [8, 9, 10, 11, 12, 13]. It should be noted here that only with the FFT-based inversion is the corresponding numerical forward Laplace transform observed. Namely, other inversion methods are one way transformation from $F(s)$ to $f(t)$. However, the FFT-based method introduces large errors and valid interval is limited to its former half. To reduce the large errors, use of acceleration methods such as Euler transformation and ε algorithm has been proposed [9, 12, 14].

In contrast to the numerical inversion, only a few methods have been proposed for numerical Laplace transformation [15, 16, 17, 18], and the FFT-based transform proposed in [18] makes a numerical Laplace transform pair with the FFT-based inversion. In the case of forward/inverse Fourier transforms, their discretization leads to the discrete forward/inverse Fourier transforms that still form an analytically reversible transform pair. The same argument, however, does not apply to the Laplace transforms: there is no discrete and reversible Laplace transform pair, and in fact, the FFT-based transform pair is not reversible exactly. Nevertheless, a numerical Laplace transform pair enables efficient analysis when one would like to move between in the time domain and in the frequency domain, for example, when analyzing frequency-dependent linear

transmission lines terminated by non-linear loads [18].

1.2 Objectives

The first objective of this thesis is to improve the handy but error prone method of the FFT-based numerical inversion of Laplace transforms aiming to extend the valid range, that is limited to the former half, to the entire interval. It further includes improvement of the FFT-based numerical Laplace transform pair in which interpolations are introduced by necessity due to the large errors.

The second objective is to implement the numerical Laplace transforms on hardware. Although an extensive number of implementations of FFT have been proposed, that of the Laplace transforms is not reported. To make possible real-time signal processing using the Laplace transforms, it is aimed to implement the Laplace transforms on an FPGA (Field Programmable Gate Array) which is a reconfigurable logic device suitable for prototyping.

The third objective is to analyze fault transients on power transmission lines and to develop a fault location method based on the analysis. By considering the simplest fault situations, it is aimed to find analytical and fundamental aspects of the fault transients.

1.3 Overview of the Thesis

This thesis is organized as follows:

Chapter 2 introduces the FFT-based numerical Laplace inversion and the FFT-based numerical Laplace transforms. It then describes the FFT-based numerical Laplace transform pair proposed in [18].

Chapter 3 presents error analysis of the FFT-based inversion and proposes another simple yet effective acceleration method. It contains numerous examples to show the effectiveness of the proposed method.

Chapter 4 gives analysis of interpolations used in the numerical Laplace transform pair and presents a clear interpretation of them. Based on the analysis, it proposes a novel computationally efficient transform pair.

Chapter 5 describes the prototyping implementation of the FFT-based numerical Laplace transforms on an FPGA.

Chapter 6 introduces analyzing method of transmission lines in the Laplace domain. It also

describes effects of lossy ground on overhead power transmission lines and some techniques for computation of matrix functions.

Chapter 7 presents expressions for fault transients on ideal transmission lines by Laguerre functions. Utilizing the expressions, a novel technique of fault location is proposed. The effects of lossy ground is also taken into account.

Finally, Chapter 8 concludes this thesis by summarizing major results obtained by the research.

Chapter 2

Fundamentals of Numerical Laplace Transforms

2.1 Introduction

This chapter describes fundamentals of numerical Laplace transforms. After giving the definition of Laplace transforms, we begin with their numerical inversions. The FFT-based method has originally been introduced by Dubner and Abate [8] in 1968 although their method uses only the Fourier cosine series. Afterwards, the so-called FFT-based method appeared in several papers [9, 10, 11, 12, 13] by using Fourier sine series as well. As pointed out in some of the papers, the FFT-based method yields better accuracy than the original one. Indeed, the valid region of obtained inversion is limited to $[0, T/2)$ for the Fourier cosine series method, and it is extended to $[0, T)$ for the FFT-based method, where the period is $2T$. Extending the valid range to the entire $[0, 2T)$ is the main purpose of Chapter 3. In the latter of this chapter, the FFT-based numerical Laplace transform is presented followed by description of the FFT-based numerical Laplace transform pair [18].

2.2 The Laplace Transform

Let $f(t)$ be a causal and real function of t, with $f(t) = 0$ for $t < 0$, the Laplace transform of $f(t)$ is defined by

$$F(s) = \mathcal{L}[f(t)](s) = \int_0^\infty f(t)\mathrm{e}^{-st}\mathrm{d}t \tag{2.1}$$

where s is the complex frequency variable. The inverse Laplace transform of the complex function $F(s)$ is defined by the Bromwich integral as

$$f(t) = \mathcal{L}^{-1}\left[F(s)\right](t) = \frac{1}{2\pi \mathrm{j}} \int_{a-\mathrm{j}\infty}^{a+\mathrm{j}\infty} F(s)\mathrm{e}^{st}\mathrm{d}s \tag{2.2}$$

where j denotes the imaginary unit $\sqrt{-1}$, and a is an arbitrary real constant such that for any singular point p of $F(s)$, $\mathrm{Re}[p] < a$ is satisfied.

We are interested here in confirming that the inversion of the Laplace transform of $f(t)$ and the Laplace transform of the inversion of $F(s)$ respectively revert to themselves. To begin with, using Eq. (2.1) in Eq. (2.2), we have

$$\begin{aligned}
\mathcal{L}^{-1}[\mathcal{L}[f(t)]] &= \frac{1}{2\pi \mathrm{j}} \int_{a-\mathrm{j}\infty}^{a+\mathrm{j}\infty} \left(\int_0^\infty f(\tau)\mathrm{e}^{-s\tau}\mathrm{d}\tau \right) \mathrm{e}^{st}\mathrm{d}s \\
&= \int_0^\infty f(\tau) \left(\frac{1}{2\pi \mathrm{j}} \int_{a-\mathrm{j}\infty}^{a+\mathrm{j}\infty} \mathrm{e}^{s(t-\tau)}\mathrm{d}s \right) \mathrm{d}\tau \\
&= \int_0^\infty f(\tau)\delta(t-\tau)\mathrm{d}\tau \\
&= \begin{cases} 0 & (t < 0) \\ \dfrac{1}{2} f(+0) & (t = 0) \\ f(t) & (t > 0) \end{cases}
\end{aligned} \tag{2.3}$$

From the second line to the third one, we used the fact that the Laplace transform of the delta function $\delta(t)$ is equal to 1. The integral in the third line generates a causal function from $f(t)$ through its integration range from $t = 0$ to $t = \infty$. In this case, the integral recovers $f(t)$ except for $f(0)$, since $f(t)$ is assumed to be a causal function.

Then, using Eq. (2.2) in Eq. (2.1), we have

$$\begin{aligned}
\mathcal{L}[\mathcal{L}^{-1}[F(s)]] &= \int_0^\infty \left(\frac{1}{2\pi \mathrm{j}} \int_{a-\mathrm{j}\infty}^{a+\mathrm{j}\infty} F(\sigma)\mathrm{e}^{\sigma t}\mathrm{d}\sigma \right) \mathrm{e}^{-st}\mathrm{d}t \\
&= \frac{1}{2\pi \mathrm{j}} \int_{a-\mathrm{j}\infty}^{a+\mathrm{j}\infty} F(\sigma) \left(\int_0^\infty \mathrm{e}^{(\sigma-s)t}\mathrm{d}t \right) \mathrm{d}\sigma \\
&= \frac{1}{2\pi \mathrm{j}} \int_{a-\mathrm{j}\infty}^{a+\mathrm{j}\infty} \frac{F(\sigma)}{s-\sigma}\mathrm{d}\sigma.
\end{aligned} \tag{2.4}$$

The last integral can be interpreted as convolution of $F(s)$ and the Laplace transform of the unit step function, $1/s$, in the s domain. Namely, it is the Hilbert transform of $F(s)$ in the s domain.

In the time domain, the Hilbert transformation corresponds to generating a causal function from $\mathcal{L}^{-1}[F(s)](t)$ in the same manner as Eq. (2.3). In this case, since $f(t)$ is a causal function, the integral gives $F(s)$.

2.3 FFT-Based Numerical Inversion

In this section, the expression for the FFT-based numerical inversion of Laplace transforms is derived. Substituting $s = a + \mathrm{j}\omega$ in Eq. (2.1), we have

$$F(a + \mathrm{j}\omega) = \int_0^\infty f(t)\mathrm{e}^{-(a+\mathrm{j}\omega)t}\mathrm{d}t \tag{2.5}$$

where a is a real constant and ω denotes the imaginary part of s. In order to discretize $F(a + \mathrm{j}\omega)$, let the sampling interval of ω be

$$\Delta\omega = \frac{2\pi}{2T} = \frac{\pi}{T} \tag{2.6}$$

where $2T$ is the time length of $f(t)$ to be computed. Then, Eq. (2.5) becomes

$$\begin{aligned} F(a + \mathrm{j}k\Delta\omega) &= \int_0^\infty f(t)\mathrm{e}^{-at}\mathrm{e}^{-\mathrm{j}k\Delta\omega t}\mathrm{d}t \\ &= \sum_{i=0}^{\infty}\int_{2iT}^{2(i+1)T} f(t)\mathrm{e}^{-at}\mathrm{e}^{-\mathrm{j}\pi kt/T}\mathrm{d}t \\ &= \int_0^{2T}\left(\sum_{i=0}^{\infty} f(t+2iT)\mathrm{e}^{-a(t+2iT)}\right)\mathrm{e}^{-\mathrm{j}\pi kt/T}\mathrm{d}t \end{aligned} \tag{2.7}$$

where k is an integer and $\mathrm{e}^{\mathrm{j}2\pi ik} = 1$ is used in the last equality. Let us define $\tilde{f}(t)$ by

$$\begin{aligned} \tilde{f}(t)\mathrm{e}^{-at} &= \sum_{i=0}^{\infty} f(t+2iT)\mathrm{e}^{-a(t+2iT)}, \\ \therefore\ \tilde{f}(t) &= f(t) + \sum_{i=1}^{\infty} f(t+2iT)e^{-2aiT}. \end{aligned} \tag{2.8}$$

The function $\tilde{f}(t)\mathrm{e}^{-at}$ is the periodization of $f(t)\mathrm{e}^{-at}$ with period $2T$, which is caused by the discretization of $F(a + \mathrm{j}\omega)$ with the interval $\Delta\omega = 2\pi/2T$. Using $\tilde{f}(t)$, Eq. (2.7) can be written as

$$F(a + \mathrm{j}k\Delta\omega) = \int_0^{2T} \tilde{f}(t)\mathrm{e}^{-at}\mathrm{e}^{-\mathrm{j}\pi kt/T}\mathrm{d}t. \tag{2.9}$$

This relation shows that $F(a + \mathrm{j}k\Delta\omega)$ is a coefficient for the Fourier series expansion for $\tilde{f}(t)\mathrm{e}^{-at}$ so that $\tilde{f}(t)\mathrm{e}^{-at}$ can be reconstructed as

$$\tilde{f}(t)\mathrm{e}^{-at} = \frac{1}{2T}\sum_{k=-\infty}^{\infty} F(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}\pi kt/T}. \tag{2.10}$$

Rewriting the above equation as

$$\tilde{f}(t) = \frac{\mathrm{e}^{at}}{2T}\sum_{k=-\infty}^{\infty} F(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}\pi kt/T} = \frac{1}{2\pi}\sum_{k=-\infty}^{\infty} F(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{(a+\mathrm{j}k\Delta\omega)t}\Delta\omega, \tag{2.11}$$

the function $\tilde{f}(t)$ can also be thought as the trapezoidal quadrature of Eq. (2.2). Truncating the infinite sum and leaving the terms for $k = -N + 1, \cdots, N - 1$, we have

$$f_N(t) = \frac{\mathrm{e}^{at}}{2T}\sum_{k=-N+1}^{N-1} F(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}\pi kt/T}. \tag{2.12}$$

By sampling $f_N(t)$ with period Δt defined by

$$\Delta t = \frac{2T}{N}, \tag{2.13}$$

the FFT-based numerical inversion of $F(s)$ is given by

$$\begin{aligned} f_N(n\Delta t) &= \frac{\mathrm{e}^{an\Delta t}}{2T}\sum_{k=-N+1}^{N-1} F(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}2\pi kn/N} \\ &= \frac{\mathrm{e}^{an\Delta t}}{T}\left\{\mathrm{Re}\left[\sum_{k=0}^{N-1} F(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}2\pi kn/N}\right] - \frac{1}{2}F(a)\right\}, \end{aligned} \tag{2.14}$$

where $n = 0, 1, \cdots, N - 1$ and the relation $F^*(s) = F(s^*)$ ($*$ denotes the complex conjugate) followed by the assumption of $f(t)$ to be a real function is used. The above numerical inversion is called 'FFT-based' since one can compute the summation in Eq. (2.14) by the FFT algorithm.

As an example, Figs. 2.1 and 2.2 show results for the unit step function $F(s) = 1/s$ and the cosine function $F(s) = s/(s^2 + \omega^2), \omega = 6\pi$ with $a = 2, 3.5, 5$, and $T = 1, N = 256$. The inversion introduces quite large errors especially for $T \le t < 2T$, and hence, only the former half ($0 \le t < T$) is acceptable. Though the amount of errors depends on the value of a and it seems to be smaller if so is a, the value of aT is actually a tradeoff between two kinds of errors as we will derive in Chapter 3. In fact, in Fig. 2.1, the line $a = 2.0$ is slightly above the exact $f(t) = 1$ even for $0 \le t < T$. That is, the offset error in the former half becomes large for small a while

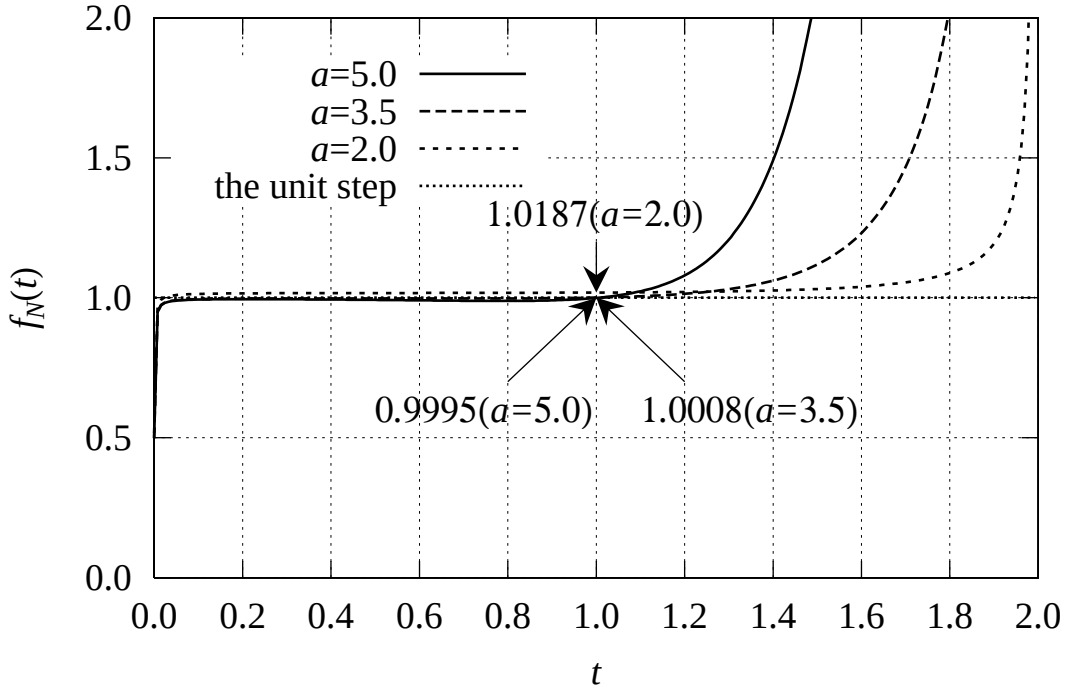


Figure 2.1: FFT-based numerical inversion of the unit step $F(s) = 1/s$. ($N = 256, T = 1$)

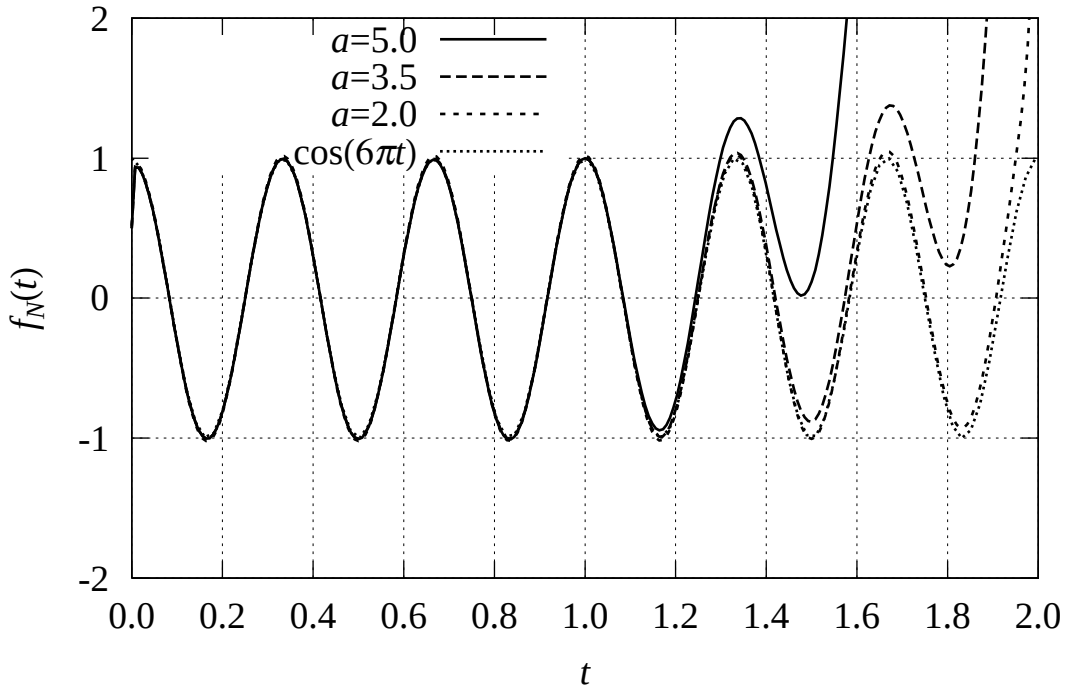


Figure 2.2: FFT-based numerical inversion of the cosine function $F(s) = s/(s^2 + \omega^2), \omega = 6\pi$ rad. ($N = 256, T = 1$)

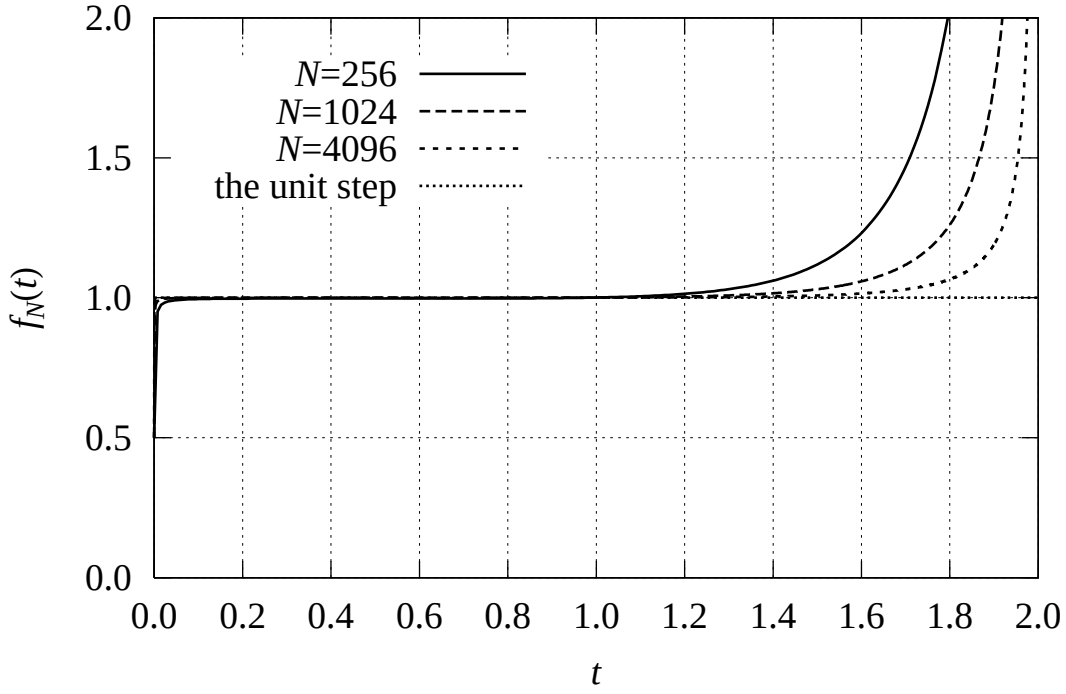


Figure 2.3: FFT-based numerical inversion of the unit step $F(s) = 1/s$. ($T = 1, a = 3.5$)

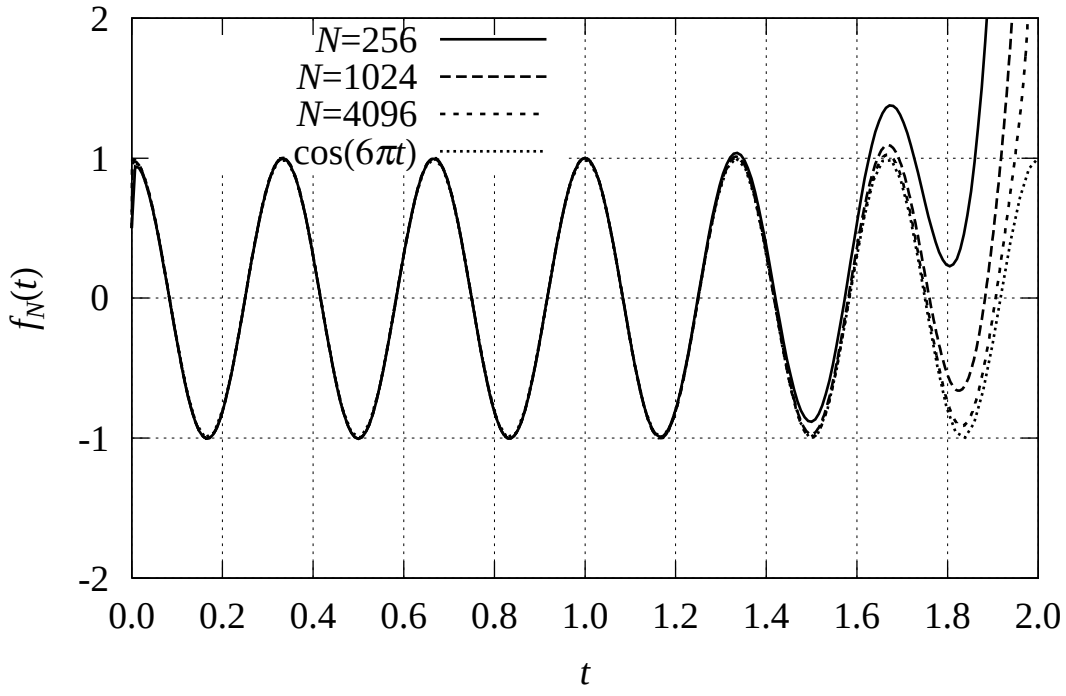


Figure 2.4: FFT-based numerical inversion of the cosine function $F(s) = s/(s^2 + \omega^2), \omega = 6\pi$. ($T = 1, a = 3.5$)

the large error in the latter half is reduced. The typical value of aT is $3 \sim 5$, and we use $aT = 3.5$ throughout this dissertation. Figures 2.3 and 2.4 present results of changing the length N for the same functions. The errors are reduced by taking the large values of N.

The FFT-based numerical inversion of Laplace transform $F(s)$ is illustratively summarized in Fig. 2.5. As the numerical inversion of N-point $F(s)$ with sampling interval $\Delta\omega$, one obtains N-point $f_N(t)$ for $0 \leq t < 2T$ with sampling period Δt. In the figure, the maximum angular frequency Ω is defined as

$$\Omega = N\Delta\omega = \frac{2\pi}{\Delta t}. \tag{2.15}$$

However, the latter half of the obtained result is discarded since it can be contaminated with large errors. Thus one obtains valid $N/2$-point $f_N(t)$ for $0 \leq t < T$. The reason why the length $2T$ is usually used instead of T lies in the rejection of the latter half.

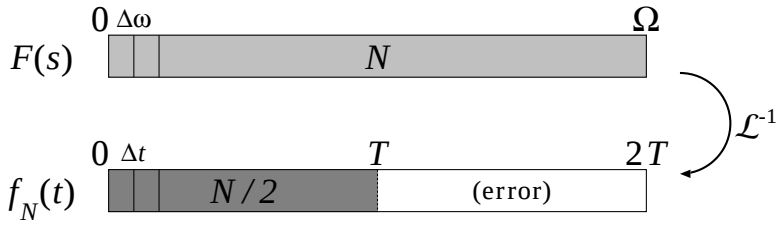


Figure 2.5: FFT-based numerical inversion of $F(s)$.

2.4 FFT-Based Numerical Transforms

This section gives the derivation of the FFT-based numerical Laplace transforms. To discretize $f(t)$, let the sampling period be

$$\Delta\tau = \frac{2\pi}{2\Omega} = \frac{\pi}{\Omega} \tag{2.16}$$

where 2Ω is the maximum angular frequency of $F(s)$ to be computed. Let $F(s)$ be the Laplace transform of $f(t)$ given by Eq. (2.1). Then Eq. (2.3) gives

$$\begin{cases} \dfrac{1}{2} f(+0) = \dfrac{1}{2\pi} \displaystyle\int_{-\infty}^{\infty} F(a + \mathrm{j}\omega)\mathrm{d}\omega \\ f(n\Delta\tau) = \dfrac{1}{2\pi} \displaystyle\int_{-\infty}^{\infty} F(a + \mathrm{j}\omega)\mathrm{e}^{(a+\mathrm{j}\omega)n\Delta\tau}\mathrm{d}\omega \quad (n \neq 0) \end{cases}. \tag{2.17}$$

For $n \neq 0$, Eq. (2.17) becomes

$$\begin{aligned} f(n\Delta\tau)\mathrm{e}^{-an\Delta\tau} &= \frac{1}{2\pi}\int_{-\infty}^{\infty} F(a+\mathrm{j}\omega)\mathrm{e}^{\mathrm{j}\omega n\Delta\tau}\mathrm{d}\omega \\ &= \frac{1}{2\pi}\sum_{i=-\infty}^{\infty}\int_{2i\Omega}^{2(i+1)\Omega} F(a+\mathrm{j}\omega)\mathrm{e}^{\mathrm{j}\pi n\omega/\Omega}\mathrm{d}\omega \\ &= \frac{1}{2\pi}\int_{0}^{2\Omega}\sum_{i=-\infty}^{\infty} F(a+\mathrm{j}(\omega+2i\Omega))\mathrm{e}^{\mathrm{j}\pi n\omega/\Omega}\mathrm{d}\omega \end{aligned} \tag{2.18}$$

where n is an integer, and $\mathrm{e}^{\mathrm{j}2\pi in} = 1$ is used in the last equality. Here, let us define $\tilde{F}(s)$ by

$$\tilde{F}(s) = \sum_{i=-\infty}^{\infty} F(s+\mathrm{j}2i\Omega). \tag{2.19}$$

The function $\tilde{F}(s)$ is the aliased function of $F(s)$ with the interval 2Ω. The aliasing is caused by the discretization of $f(t)$ with the period $\Delta\tau = 2\pi/2\Omega$. Using $\tilde{F}(s)$, Eq. (2.18) can be written as

$$f(n\Delta\tau)\mathrm{e}^{-an\Delta\tau} = \frac{1}{2\pi}\int_{0}^{2\Omega}\tilde{F}(a+\mathrm{j}\omega)\mathrm{e}^{\mathrm{j}\pi n\omega/\Omega}\mathrm{d}\omega. \tag{2.20}$$

In the same way, for $n = 0$, Eq. (2.17) leads to

$$\frac{1}{2}f(+0) = \frac{1}{2\pi}\int_{0}^{2\Omega}\tilde{F}(a+\mathrm{j}\omega)\mathrm{d}\omega. \tag{2.21}$$

These relations show that $f(+0)/2$, $f(n\Delta\tau)\mathrm{e}^{-an\Delta\tau}$ is a coefficient for the Fourier series expansion for $\tilde{F}(a+\mathrm{j}\omega)$ with respect to ω so that $\tilde{F}(a+\mathrm{j}\omega)$ can be reconstructed as

$$\tilde{F}(a+\mathrm{j}\omega) = \frac{\pi}{\Omega}\left[\frac{1}{2}f(+0) + \sum_{n=1}^{\infty} f(n\Delta\tau)\mathrm{e}^{-an\Delta\tau}\mathrm{e}^{-\mathrm{j}\pi n\omega/\Omega}\right] \tag{2.22}$$

where the causality, $f(n\Delta\tau) = 0$ for $n < 0$, is used. By rewriting $\tilde{F}(a+\mathrm{j}\omega)$ as

$$\tilde{F}(a+\mathrm{j}\omega) = \left[\frac{1}{2}f(+0) + \sum_{n=1}^{\infty} f(n\Delta\tau)\mathrm{e}^{-(a+\mathrm{j}\omega)n\Delta\tau}\right]\Delta\tau, \tag{2.23}$$

the function $\tilde{F}(a+\mathrm{j}\omega)$ turns out to be the trapezoidal quadrature of Eq. (2.1). Truncating the infinite sum of Eq. (2.22) at $n = N$, we have

$$F_N(a+\mathrm{j}\omega) = \frac{\pi}{\Omega}\left[\frac{1}{2}f(+0) + \sum_{n=1}^{N-1} f(n\Delta\tau)\mathrm{e}^{-an\Delta\tau}\mathrm{e}^{-\mathrm{j}\pi n\omega/\Omega}\right], \tag{2.24}$$

and by sampling $F_N(a + \mathrm{j}\omega)$ with interval $\Delta\nu$ defined by

$$\Delta\nu = \frac{2\Omega}{N}, \tag{2.25}$$

the FFT-based numerical Laplace transform of $f(t)$ is given by

$$F_N(a + \mathrm{j}k\Delta\nu) = \frac{\pi}{\Omega}\left[\frac{1}{2}f(+0) + \sum_{n=1}^{N-1} f(n\Delta\tau)\mathrm{e}^{-an\Delta\tau}\mathrm{e}^{-\mathrm{j}2\pi kn/N}\right], \tag{2.26}$$

where $k = 0, 1, \cdots, N-1$. The summation in Eq. (2.26) can also be computed by the FFT algorithm.

The FFT-based numerical Laplace transform of $f(t)$ is illustratively summarized in Fig. 2.6. The time length T is defined as

$$T = N\Delta\tau = \frac{2\pi}{\Delta\nu}. \tag{2.27}$$

The latter half of the result is ignored since it is the aliasing introduced in Eq. (2.19), and at the same time, it is the complex conjugate of the reversed former half as shown by the following identities:

$$F_N^*(a + \mathrm{j}k\Delta\nu) = F_N(a - \mathrm{j}k\Delta\nu) = F_N(a + \mathrm{j}(N-k)\Delta\nu). \tag{2.28}$$

Therefore one obtains $N/2$-point $F_N(a + \mathrm{j}\omega)$ for $0 \le \omega < \Omega$ with sampling interval $\Delta\nu$.

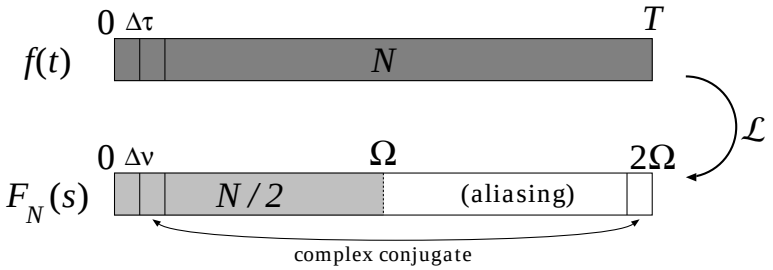


Figure 2.6: FFT-based numerical Laplace transform of $f(t)$.

2.5 Numerical Laplace Transform Pair

In this section, the numerical inverse/forward Laplace transforms described in the previous sections are combined to form a numerical Laplace transform pair. The number of data, however,

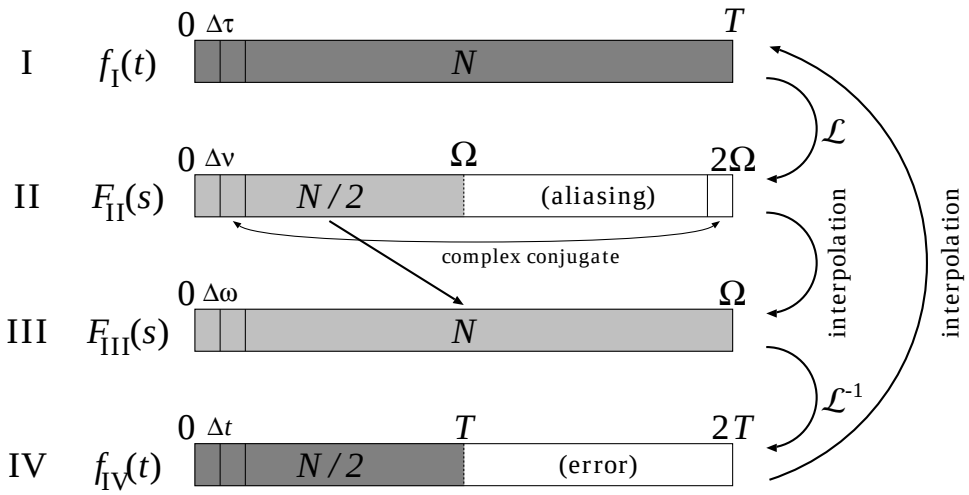


Figure 2.7: FFT-based numerical Laplace transform pair.

is halved after both inverse and forward transformations due to the large errors and the aliasing effect, respectively. Hence, the direct application of the both transformations in turn results in exponentially decreasing number of data points. This difficulty was remedied by interpolating the results after each inverse/forward transformation [18].

Figure 2.7 illustrates the numerical Laplace transform pair. The step sizes are related by

$$\Delta t = 2\Delta\tau = \frac{2T}{N} = \frac{2\pi}{\Omega}, \quad \Delta\nu = 2\Delta\omega = \frac{2\Omega}{N} = \frac{2\pi}{T} \tag{2.29}$$

Based on Eq. (2.26), the numerical Laplace transform of $f_{\mathrm{I}}(n\Delta\tau)$ is computed by

$$\begin{aligned} F_{\mathrm{II}}(a + \mathrm{j}k\Delta\nu) &= \frac{\pi}{\Omega}\left[\frac{1}{2}f_{\mathrm{I}}(0) + \sum_{n=1}^{N-1} f_{\mathrm{I}}(n\Delta\tau)\mathrm{e}^{-an\Delta\tau}\mathrm{e}^{-\mathrm{j}2\pi kn/N}\right] \\ &= \frac{\pi}{\Omega}\left[\sum_{n=0}^{N-1} f_{\mathrm{I}}(n\Delta\tau)\mathrm{e}^{-an\Delta\tau}\mathrm{e}^{-\mathrm{j}2\pi kn/N} - \frac{1}{2}f_{\mathrm{I}}(0)\right], \end{aligned} \tag{2.30}$$

where we identified $f_{\mathrm{I}}(+0)$ with $f_{\mathrm{I}}(0)$ since $f_{\mathrm{I}}(n\Delta\tau)$ is a discrete function. Subsequently, $F_{\mathrm{II}}(a + \mathrm{j}k\Delta\nu)$ is interpolated into $F_{\mathrm{III}}(a+\mathrm{j}k\Delta\omega)$ as follows. For even points, the function values are simply taken from $F_{\mathrm{II}}(a + \mathrm{j}k\Delta\nu)$ as

$$F_{\mathrm{III}}(a + \mathrm{j}2k\Delta\omega) = F_{\mathrm{II}}(a + \mathrm{j}k\Delta\nu), \quad k = 0, 1, \cdots, \frac{N}{2} - 1. \tag{2.31}$$

Then, for odd points, they are computed by using Eq. (2.24) as

$$F_{\text{III}}(a+\mathrm{j}(2k+1)\Delta\omega) = \frac{\pi}{\Omega}\left[\sum_{n=0}^{N-1} f_{\text{I}}(n\Delta\tau)\mathrm{e}^{-an\Delta\tau}\mathrm{e}^{-\mathrm{j}\pi n(2k+1)\Delta\omega/\Omega} - \frac{1}{2}f_{\text{I}}(0)\right]$$
$$= \frac{\pi}{\Omega}\left[\sum_{n=0}^{N-1}\left(f_{\text{I}}(n\Delta\tau)\mathrm{e}^{-\mathrm{j}\pi n/N}\right)\mathrm{e}^{-an\Delta\tau}\mathrm{e}^{-\mathrm{j}2\pi nk/N} - \frac{1}{2}f_{\text{I}}(0)\right], \qquad (2.32)$$

where $k = 0, 1, \cdots, N/2-1$. Referring to Eq. (2.30), the odd data points can be computed as the numerical transform of $f_{\text{I}}(n\Delta\tau)\mathrm{e}^{-\mathrm{j}\pi n/N}$.

Likewise, based on Eq. (2.14), the inversion of $F_{\text{III}}(a+\mathrm{j}k\Delta\omega)$ is computed by

$$f_{\text{IV}}(n\Delta t) = \frac{\mathrm{e}^{an\Delta t}}{T}\left\{\mathrm{Re}\left[\sum_{k=0}^{N-1} F_{\text{III}}(a+\mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}2\pi kn/N}\right] - \frac{1}{2}F_{\text{III}}(a)\right\}. \qquad (2.33)$$

The interpolation of $f_{\text{IV}}(n\Delta t)$ into $f_{\text{I}}(n\Delta\tau)$ may be computed by using $F_{\text{III}}(a+\mathrm{j}k\Delta\omega)$ in the same manner as that of $F_{\text{II}}(a+\mathrm{j}k\Delta\nu)$. That is, for even points, the values of $f_1(2n\Delta\tau)$ are simply taken from $f_{\text{IV}}(n\Delta t)$ as

$$f_{\text{I}}(2n\Delta\tau) = f_{\text{IV}}(n\Delta t), \quad n = 0, 1, \cdots, \frac{N}{2}-1, \qquad (2.34)$$

and for odd points, they are computed by using Eq. (2.12) as

$$f_{\text{I}}((2n+1)\Delta\tau) = \frac{\mathrm{e}^{a(2n+1)\Delta\tau}}{2T}\sum_{k=-N+1}^{N-1} F_{\text{III}}(a+\mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}\pi k(2n+1)\Delta\tau/T}$$
$$= \mathrm{e}^{a\Delta\tau}\frac{\mathrm{e}^{an\Delta t}}{2T}\sum_{k=-N+1}^{N-1}\left(F_{\text{III}}(a+\mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}\pi k/N}\right)\mathrm{e}^{\mathrm{j}2\pi kn/N}$$
$$= \mathrm{e}^{a\Delta\tau}\frac{\mathrm{e}^{an\Delta t}}{T}\left\{\mathrm{Re}\left[\sum_{k=0}^{N-1}\left(F_{\text{III}}(a+\mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}\pi k/N}\right)\mathrm{e}^{\mathrm{j}2\pi kn/N}\right] - \frac{1}{2}F_{\text{III}}(a)\right\}, \qquad (2.35)$$

where $n = 0, 1, \cdots, N/2-1$. Referring to Eq. (2.33), the odd data points can be computed as the numerical inversion of $F_{\text{III}}(a+\mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}\pi k/N}$ multiplied by $\mathrm{e}^{a\Delta\tau}$.

In Fig. 2.8, the solid line shows the interpolation of numerical inversion of the unit step $F_{\text{III}}(s) = 1/s$. The oscillation originates from the lack of bandwidth in $F_{\text{III}}(a+\mathrm{j}k\Delta\omega)$. Although the bandwidth of $[0, 2\Omega)$ is needed to obtain $f(t)$ with sampling period $\Delta\tau = 2\pi/2\Omega$, $F_{\text{III}}(a+\mathrm{j}k\Delta\omega)$ merely covers $[0, \Omega)$, which is the half of the necessary bandwidth. Accordingly, $f_{\text{IV}}(n\Delta t)$ is usually interpolated in the time domain by using linear interpolation or splines, instead of using

$F_{\mathrm{III}}(s)$, to avoid the oscillations. The dashed line in Fig. 2.8 shows linear interpolation of $f_{\mathrm{IV}}(n\Delta t)$ defined by

$$\begin{cases} f_{\mathrm{I}}((2n+1)\Delta\tau) = \dfrac{f_{\mathrm{IV}}(n\Delta t) + f_{\mathrm{IV}}((n+1)\Delta t)}{2} & \left(n = 0, 1, \cdots, \dfrac{N}{2} - 2\right) \\ f_{\mathrm{I}}((N-1)\Delta\tau) = f_{\mathrm{I}}((N-2)\Delta\tau) = f_{\mathrm{IV}}\left(\left(\dfrac{N}{2} - 1\right)\Delta t\right) & \left(n = \dfrac{N}{2} - 1\right) \end{cases}, \quad (2.36)$$

where the value of the last point ($n = N/2 - 1$) is simply taken to be the previous one.

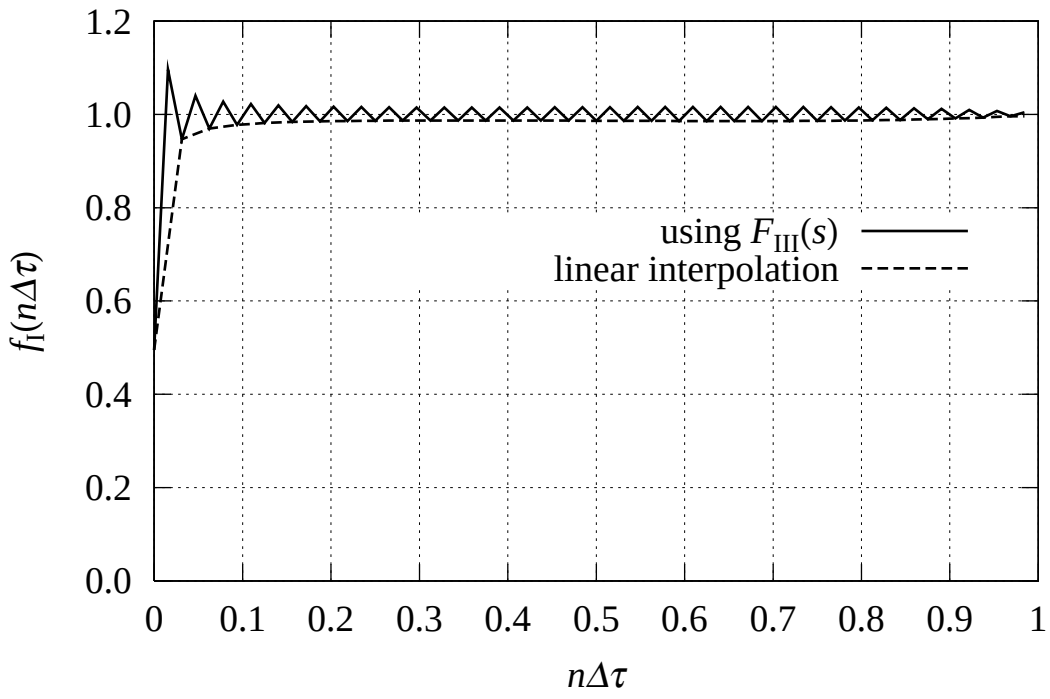


Figure 2.8: Interpolation of $f_{\mathrm{IV}}(n\Delta t)$. ($N = 64, T = 1, a = 3.5$)

Chapter 3

Acceleration of Convergence of FFT-Based Inversion

3.1 Introduction

As shown in Section 2.3, the FFT-based numerical Laplace inversion introduces large errors especially for the latter half of the result. Namely, the valid region of the obtained result is $[0, T)$ where the period is $2T$. In order to reduce the errors, Durbin [9] proposed extension of the summation length in Eq.(2.14) by replacing the upper limit $N-1$ by $K \cdot N - 1$, where K is $10 \sim 100$. This remedy is equivalent to taking the number of sampling points to be $K \cdot N$ while the actual number of sampling points is left unchanged and so is the FFT length. Although this modification evidently reduces the errors as seen from Figs. 2.3 and 2.4, it is a time-consuming task since it requires evaluation of $F(s)$ for K-times as many values of s.

In this chapter, we first analyze errors introduced by the FFT-based inversion, and then, propose another simple yet effective method to reduce the errors and extend the valid region to the entire $[0, 2T)$. The comparison with the method mentioned above is also presented.

3.2 Error Analysis of the FFT-Based Inversion

Let us first show two more examples of the FFT-based inversion in Figs. 3.1 and 3.2, which are respectively the numerical inversion of the sine function and $F(s) = 1/\sqrt{s}$ computed by Eq. (2.14). In contrast to the examples presented in Section 2.3, the inversion of the sine function is successfully obtained for almost the entire region without errors. On the other hand, $F(s) = 1/\sqrt{s}$ is one of the functions totally difficult to invert by the FFT-based method. In this section,

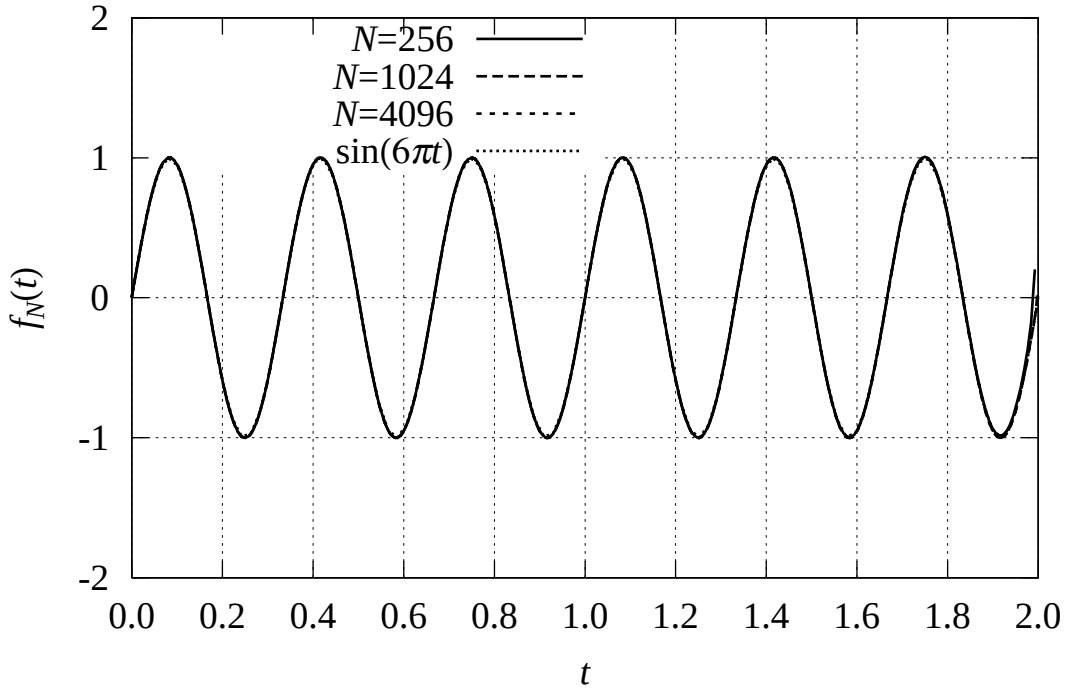


Figure 3.1: FFT-based numerical inversion of the sine function $F(s) = \omega/(s^2 + \omega^2), \omega = 6\pi$ rad. ($T = 1, a = 3.5$. The four lines overlap each other.)

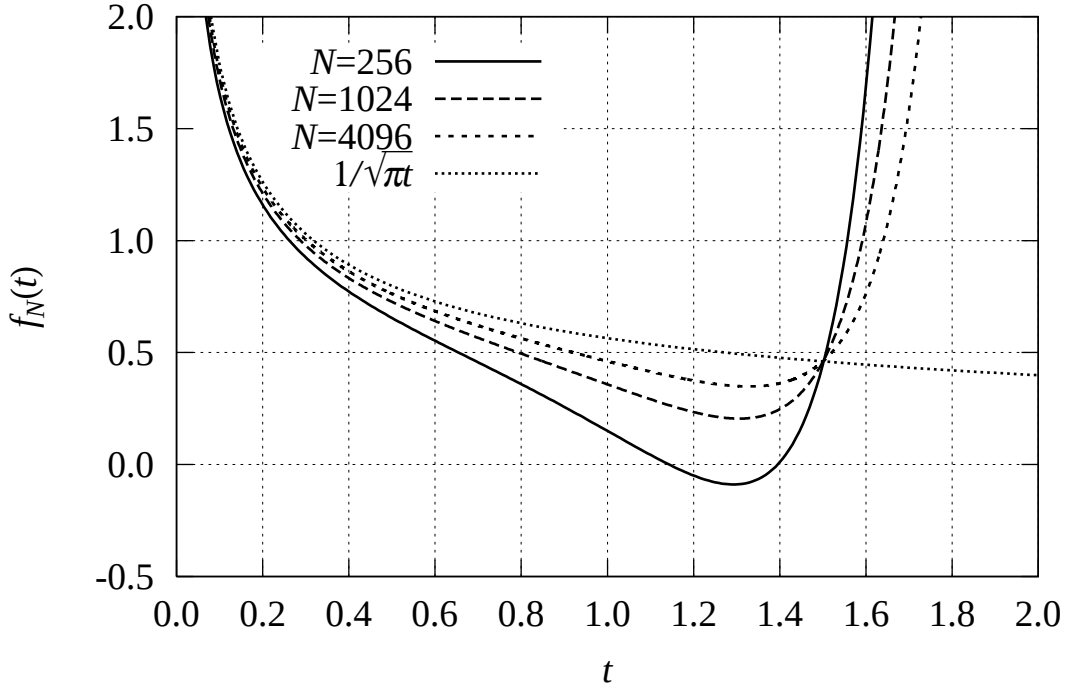


Figure 3.2: FFT-based numerical inversion of $F(s) = 1/\sqrt{s}$, $\mathcal{L}^{-1}[F](t) = 1/\sqrt{\pi t}$. ($T = 1, a = 3.5$)

we clarify what kind of nature of those Laplace transforms brings about such differences.

The FFT-based numerical inversion of Laplace transforms introduces two kinds of errors except for the computational round off errors as derived in Section 2.3. That is,

$$f_N(t) = f(t) + e_1(t) + e_2(t) \tag{3.1}$$

$$e_1(t) = \tilde{f}(t) - f(t) \tag{3.2}$$

$$e_2(t) = f_N(t) - \tilde{f}(t) \tag{3.3}$$

where $e_1(t)$ is the discretization error due to the application of trapezoidal quadrature to Eq. (2.2), and $e_2(t)$ is the truncation error originating from the truncation of the resulting infinite sum.

The discretization error $e_1(t)$ is given by Eq. (2.8) as

$$e_1(t) = \sum_{i=1}^{\infty} f(t + 2iT)\mathrm{e}^{-2aiT}. \tag{3.4}$$

Supposing that $f(t)$ is bounded by a constant C as $|f(t)| \leq C$, then we have

$$|e_1(t)| = \left| \sum_{i=1}^{\infty} f(t + 2iT)\mathrm{e}^{-2aiT} \right| \leq \sum_{i=1}^{\infty} |f(t + 2iT)|\,\mathrm{e}^{-2aiT} \leq \sum_{i=1}^{\infty} Ce^{-2aiT}$$

$$\therefore\ |e_1(t)| \leq \frac{C}{\mathrm{e}^{2aT} - 1}. \tag{3.5}$$

Since a value about $3 \sim 5$ is usually chosen for aT, the above error bound is small enough. For example, with $aT = 3.5$:

$$|e_1(t)| \leq \frac{C}{\mathrm{e}^{7} - 1} = 0.000912C. \tag{3.6}$$

Hence, the discretization error does not contribute to the large errors in the examples shown in Section 2.3 since $C = 1$ for those examples. Instead, the offset error seen in Fig. 2.1 for $a = 2$ stems from the discretization error. In the case of the unit step function $f(t) = 1$, the error is analytically given by

$$e_1(t) = \frac{1}{\mathrm{e}^{2aT} - 1}, \tag{3.7}$$

and is 0.0187 for $a = 2, T = 1$, that coincides with Fig. 2.1.

Then, the truncation error $e_2(t)$ is given by Eq. (2.12) as

$$e_2(t) = \frac{\mathrm{e}^{at}}{2T} \sum_{|k| \geq N} F(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}\pi kt/T}. \tag{3.8}$$

The truncation error gives rise to the large errors shown in the examples, and in fact, the above expression explains their exponential increase in the latter half.

In order to evaluate $e_2(t)$, let us define a function $E_2(t)$ as the absolute summation of $e_2(t)$ by

$$E_2(t) = \frac{\mathrm{e}^{at}}{2T} \sum_{|k|\geq N} \left| F(a + \mathrm{j}k\Delta\omega) \right|. \tag{3.9}$$

Supposing that $F(s) = 1/s^m$ (m is a positive real number), $E_2(t)$ is then given by

$$E_2(t) = \frac{\mathrm{e}^{at}}{2T} \cdot 2 \sum_{k=N}^{\infty} \left| F(a + \mathrm{j}k\Delta\omega) \right| = \frac{\mathrm{e}^{at}}{T} \sum_{k=N}^{\infty} \left| \frac{1}{(a + \mathrm{j}k\Delta\omega)^m} \right|, \tag{3.10}$$

where we used the relation $|F(a - \mathrm{j}k\Delta\omega)| = |F^*(a + \mathrm{j}k\Delta\omega)| = |F(a + \mathrm{j}k\Delta\omega)|$. From the inequality

$$\frac{1}{a + k\Delta\omega} < \frac{1}{|a + \mathrm{j}k\Delta\omega|} < \frac{1}{\Delta\omega} \cdot \frac{1}{k} \quad (k \geq N), \tag{3.11}$$

$E_2(t)$ is bounded as

$$\frac{\mathrm{e}^{at}}{T(\Delta\omega)^m} \sum_{k=N}^{\infty} \frac{1}{(k + a/\Delta\omega)^m} < E_2(t) < \frac{\mathrm{e}^{at}}{T(\Delta\omega)^m} \sum_{k=N}^{\infty} \frac{1}{k^m}. \tag{3.12}$$

In the case of $m \leq 1$, the lower bound for $E_2(t)$, i.e. the left hand side of Eq. (3.12), does not converge, and thereby $e_2(t)$ is not absolutely convergent. This fact implies the quite large errors for $m = 1$ as shown in Figs. 2.1~2.4, and for $m = 1/2$ in Fig. 3.2. On the contrary, in the case of $m > 1$, the upper bound for $E_2(t)$, i.e. the right hand side of Eq. (3.12), is absolutely convergent. Therefore the truncation error $e_2(t)$ for $m > 1$ is markedly smaller than that for $m \leq 1$. For example, the inversion of the sine function, for which $m = 2$ for $s \to \infty$, is not contaminated by the large errors as shown in Fig. 3.1.

Moreover, from the elementary inequality

$$\sum_{k=N}^{\infty} \frac{1}{k^m} < \frac{1}{(m-1)(N-1)^{m-1}} \quad (m > 1), \tag{3.13}$$

we have

$$|e_2(t)| \leq E_2(t) < \frac{\mathrm{e}^{at}}{T(\Delta\omega)^m} \cdot \frac{1}{(m-1)(N-1)^{m-1}} = \frac{\mathrm{e}^{at}}{\pi(m-1)} \left(\frac{T}{\pi(N-1)} \right)^{m-1}. \tag{3.14}$$

Provided that $T < \pi(N-1)$, the above upper bound becomes smaller for larger value of m. In other words, if m is large, high-frequency components of $1/s^m$ are small and so is the truncation error. The same argument will be concluded if $F(s) = O(1/s^m)$ for $s \to \infty$, since the error is composed of the truncated high frequency components as defined by Eq. (3.8) so that the behavior of $F(s)$ at $s \to \infty$ dominates the truncation error.

3.3 Acceleration of Convergence

3.3.1 Acceleration Method

The error analysis in the previous section found out that a Laplace transform $F(s) = O(1/s^m)$, $s \to \infty$ can be inverted without the large errors by the FFT-based method when $m > 1$. Based on this result, we consider of transforming a Laplace transform $F(s)$ to another Laplace transform $G(s)$ which the FFT-based method will invert with better accuracy than the original, and after the inversion, transforming back the obtained inversion $g(t)$, in the time domain, to the desired function $f(t)$ by the method corresponding to the first transformation in the s domain.

As such a transformation, we recall here that the complex frequency variable s acts as the differential operator in the time domain, that is, the following identity holds:

$$f(t) = \mathcal{L}^{-1}[F(s)] = \mathcal{L}^{-1}\left[s^i \cdot \frac{F(s)}{s^i}\right] = \frac{\mathrm{d}^i}{\mathrm{d}t^i}\mathcal{L}^{-1}\left[\frac{F(s)}{s^i}\right]. \tag{3.15}$$

Since the factor $1/s^i$ suppresses the high-frequency components of $F(s)$, the truncation error in the inversion of $F(s)/s^i$ will be reduced. This idea is best shown in Figs. 2.4 and 3.1. The Laplace transform of the sine function is that of the cosine function multiplied by ω/s, so that they give an example of the idea with $i = 1$. The inversion of the cosine function can be obtained with better accuracy as the numerical differentiation of the inverted sine function.

Putting it all together, our proposal is to compute the inversion of $F(s)$ in the following steps:

1. Calculate $G(s) = \dfrac{F(s)}{s^i}$.
2. Invert $G(s)$ by the FFT-based method to obtain $g_N(n\Delta t)$ for $n = 0, \cdots, N-1$. The function $g_N(t)$ is defined similarly to Eq. (2.12) by

$$g_N(t) = \frac{\mathrm{e}^{at}}{2T}\sum_{k=-N+1}^{N-1} G(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}\pi kt/T}. \tag{3.16}$$

3. Compute the numerical i-th differential of $g_N(n\Delta t)$ by

$$\hat{f}_N(n\Delta t) = \frac{1}{(\Delta t)^i}\Delta^i g_N(n\Delta t), \tag{3.17}$$

where $\hat{f}_N(n\Delta t)$ is the proposed numerical inversion and Δ^i denotes the i-th difference operator with respect to n.

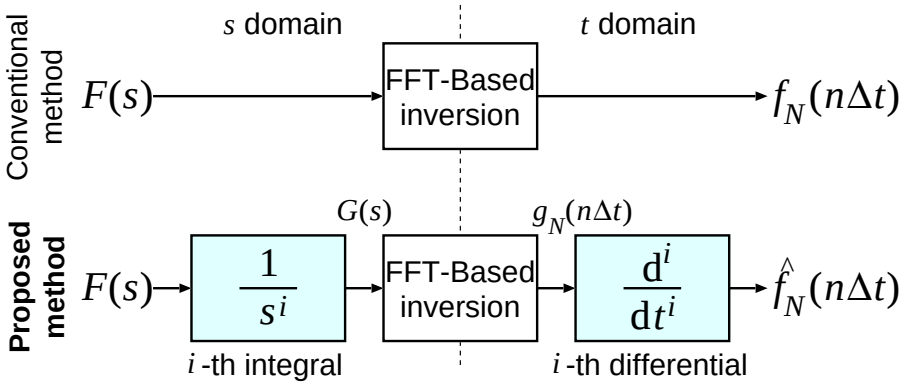


Figure 3.3: Procedure diagram of the proposed numerical Laplace inversion.

For the i-th difference operation, we use the central difference defined for even i by

$$\Delta^i g_N(n\Delta t) = \sum_{j=0}^{i} (-1)^j {}_iC_j g_N((n + i/2 - j)\Delta t), \tag{3.18}$$

where ${}_iC_j$ is the binomial coefficient, and for odd i by

$$\Delta^i g_N(n\Delta t) = \frac{1}{2}\Delta^{i-1}\left[g_N((n+1)\Delta t) - g_N((n-1)\Delta t)\right]. \tag{3.19}$$

These formulae, however, are not applicable to end points such as $n = 0$ and $n = N - 1$. For those end points, we develop the second order numerical i-th differential as well. For example with $i = 2$, the end points are computed as follows:

$$\Delta^2 g_N(0\Delta t) = 2g_N(0\Delta t) - 5g_N(1\Delta t) + 4g_N(2\Delta t) - g_N(3\Delta t) \tag{3.20}$$

$$\begin{aligned}\Delta^2 g_N((N-1)\Delta t) = &-g_N((N-1)\Delta t) + 4g_N((N-2)\Delta t) \\ &- 5g_N((N-3)\Delta t) + 2g_N((N-4)\Delta t)\end{aligned} \tag{3.21}$$

Figures 3.4 and 3.5 show results of the proposed inversion applied to the unit step function with $N = 256, T = 1, a = 3.5$. Note that it reverts to the conventional FFT-based method when $i = 0$. In Fig. 3.4, the error decreases along with increasing of i, and $i = 4$ gives remarkably good result. For $i = 8$ in Fig. 3.5, however, the result is completely contaminated by the round-off errors that stem from the numerical differentiation. Therefore, the value of i is a tradeoff between the truncation error and the round-off errors.

Figure 3.6 illustrates maximum absolute error, $\max_t \left|\hat{f}_N(t) - f(t)\right|$, for the step function $F(s) = 1/s$ with $i = 0, 1, \cdots, 8$. The error decreases along with i up to $i = 4$ for the moderate number

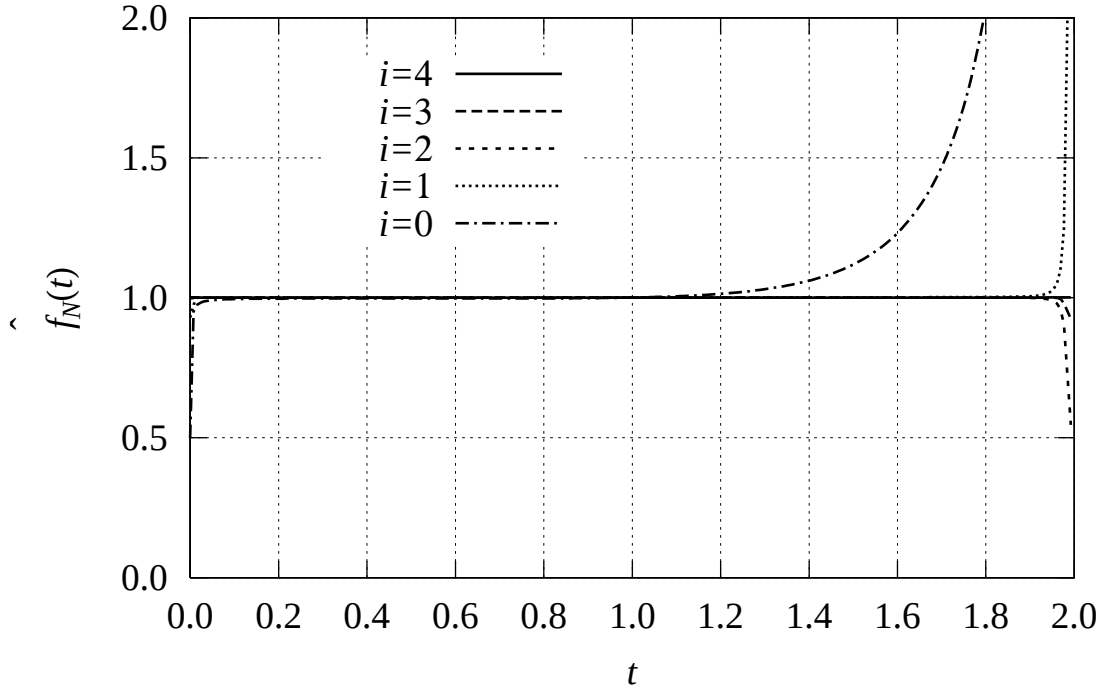


Figure 3.4: Proposed numerical inversion of the step function $F(s) = 1/s$. ($N = 256, T = 1, a = 3.5, i = 0, 1, \cdots, 4$)

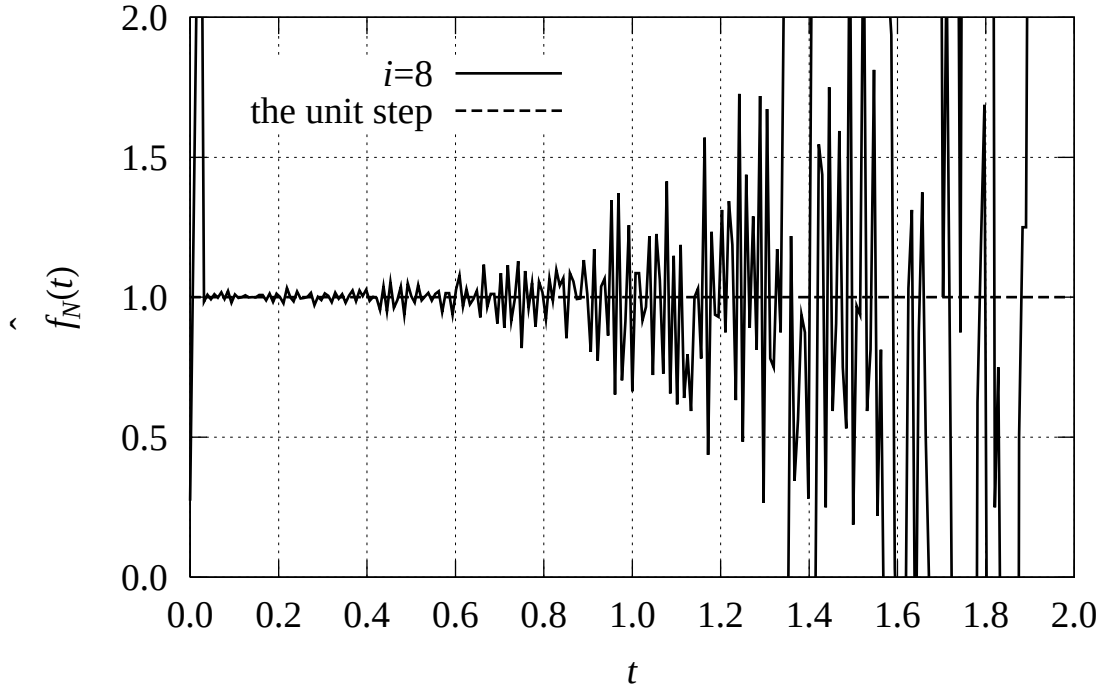


Figure 3.5: Proposed numerical inversion of the step function $F(s) = 1/s$. ($N = 256, T = 1, a = 3.5, i = 8$)

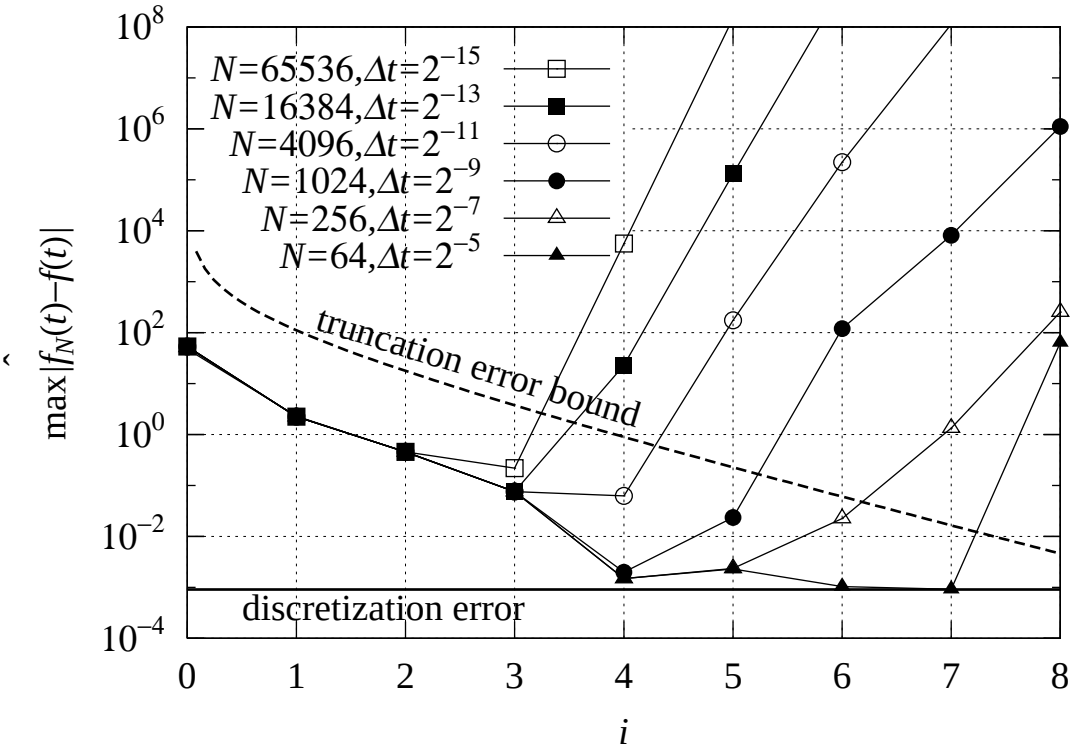


Figure 3.6: Maximum absolute errors of the proposed method for $F(s) = 1/s$.

of points $N \leq 1024$, and up to $i = 3$ for $N > 1024$. For i more than 4 or 3, the errors increase and they are mostly the round-off errors originating from the numerical i-th differentiation.

3.3.2 Error Analysis of the Proposed Method

In this section, we analyse errors involved by the proposed method by restricting ourselves specifically to $F(s) = 1/s$. For the unit step function, we have

$$g(t) = \mathcal{L}^{-1}[G(s)] = \mathcal{L}^{-1}\left[\frac{1}{s^{i+1}}\right] = \frac{t^i}{i!}, \tag{3.22}$$

$$g_N(t) = g(t) + e_1(t) + e_2(t), \tag{3.23}$$

$$e_1(t) = \sum_{j=1}^{\infty} \frac{(t + 2jT)^i}{i!} \mathrm{e}^{-2ajT}, \tag{3.24}$$

$$e_2(t) = \frac{\mathrm{e}^{at}}{2T} \sum_{|k|\geq N} \frac{1}{(a + \mathrm{j}k\Delta\omega)^{i+1}} \mathrm{e}^{\mathrm{j}\pi kt/T}. \tag{3.25}$$

Since $g(t)$ and $e_1(t)$ are monotonic smooth functions between the sampling points $t = n\Delta t$, the i-th difference of them can be approximated by the i-th differential as

$$\frac{1}{(\Delta t)^i}\Delta^i g(n\Delta t) = \frac{\mathrm{d}^i}{\mathrm{d}t^i} g(n\Delta t) + O\left((\Delta t)^2\right) = f(n\Delta t) + O\left((\Delta t)^2\right), \tag{3.26}$$

$$\frac{1}{(\Delta t)^i}\Delta^i e_1(n\Delta t) = \frac{\mathrm{d}^i}{\mathrm{d}t^i} e_1(n\Delta t) + O\left((\Delta t)^2\right) = \frac{1}{\mathrm{e}^{2aT} - 1} + O\left((\Delta t)^2\right). \tag{3.27}$$

Thus, together with Eqs. (3.17) and (3.23), we have

$$\begin{aligned}\hat{f}_N(n\Delta t) &= \frac{1}{(\Delta t)^i}\Delta^i g_N(n\Delta t) = \frac{1}{(\Delta t)^i}\Delta^i \left(g(n\Delta t) + e_1(n\Delta t) + e_2(n\Delta t)\right) \\ &= f(n\Delta t) + \frac{1}{\mathrm{e}^{2aT} - 1} + \frac{1}{(\Delta t)^i}\Delta^i e_2(n\Delta t) + O\left((\Delta t)^2\right).\end{aligned} \tag{3.28}$$

On the other hand, the truncation error $e_2(t)$ is an oscillatory function between the sampling points as shown in Fig. 3.7 since it is composed of the truncated high frequency components. Hence, the i-th difference, $\Delta^i e_2(n\Delta t)$, cannot be approximated by the i-th differential. Instead, we evaluate $\Delta^i e_2(n\Delta t)$ by using Eqs. (3.18) and (3.19). Since, in those equations, $\Delta^i e_2(n\Delta t)$ is given by less than 2^i times additions/subtractions of $e_2(n\Delta t)$, it is bounded as

$$\left|\frac{1}{(\Delta t)^i}\Delta^i e_2(n\Delta t)\right| \le \frac{2^i}{(\Delta t)^i} \max_n |e_2(n\Delta t)|. \tag{3.29}$$

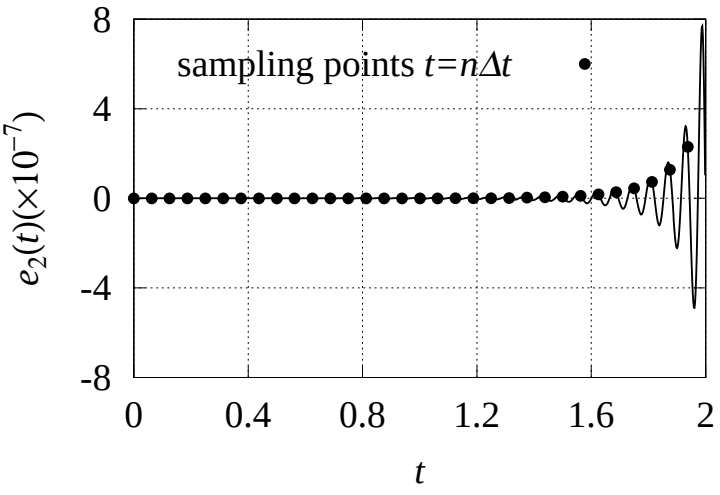


Figure 3.7: The truncation error $e_2(t)$ for $F(s) = 1/s, i = 4, N = 16$. $(T = 1, a = 3.5)$

Using Eq. (3.25) and Eq. (3.14) with $m = i + 1$, we have

$$\frac{2^i}{(\Delta t)^i}\max_n |e_2(n\Delta t)| < \frac{2^i}{(\Delta t)^i}\cdot\frac{\mathrm{e}^{2aT}}{\pi i(1-1/N)^i}\left(\frac{\Delta t}{2\pi}\right)^i$$
$$= \frac{\mathrm{e}^{2aT}}{i\pi^{i+1}(1-1/N)^i} = \frac{\mathrm{e}^{2aT}}{i\pi^{i+1}}\left(1+O\left(\frac{1}{N}\right)\right) \tag{3.30}$$

$$\therefore \quad \left|\frac{1}{(\Delta t)^i}\Delta^i e_2(n\Delta t)\right| < \frac{\mathrm{e}^{2aT}}{i\pi^{i+1}}\left(1+O\left(\frac{1}{N}\right)\right). \tag{3.31}$$

In Fig. 3.6, the solid line shows the approximated discretization error, $1/(\mathrm{e}^{2aT}-1)$, given by Eq. (3.27), and the dashed line gives the upper bound of the truncation error, $\mathrm{e}^{2aT}/(i\pi^{i+1})$, derived above. Although the truncation error bound is an overestimation, it well captures the decreasing of the errors along with i.

3.4 Numerical Examples

Let us look closely at Fig. 3.4 first. For $i = 4$, the maximum error is 0.00149 at $n = N - 1 = 255$. With relative error tolerance of 0.5%, the proposed inversion is valid for the entire region $0 \le t < 2$, while the FFT-based inversion is valid for only $0.09 < t < 1.09$.

Figures 3.8~3.11 present the proposed numerical inversion of the cosine function, the sine function, $F(s) = 1/\sqrt{s}$, and the square wave. Note again that the lines $i = 0$ are identical to the numerical inversions by the conventional FFT-based method. The truncation errors are remarkably reduced so that the lines $i = 4$ and the analytical ones almost overlap each other. In addition, Fig. 3.10 shows that the proposed method enables inversion of Laplace transforms that the FFT-based method cannot invert, although several end points still contain errors. Figure 3.11 shows another benefit of using the proposed method. In the figure, not only the truncation error is reduced, but the Gibbs phenomenon, which arises at discontinuous points, is also avoided. The discontinuities are removed by the i-th integration, i.e. the division by s^i, and thereby the overshoots disappeared.

In order to see the wide applicability of the proposed method, we compare it with the conventional FFT-based method on various types of functions used in [1]. The accuracy of a numerical inversion is measured in its root-mean-square deviation defined by

$$L_N = \left[\frac{1}{N-1}\sum_{n=1}^{N-1}(f_N(n\Delta t) - f(n\Delta t))^2\right]^{1/2}, \tag{3.32}$$

$$\hat{L}_N = \left[\frac{1}{N-1} \sum_{n=1}^{N-1} \left(\hat{f}_N(n\Delta t) - f(n\Delta t) \right)^2 \right]^{1/2}, \tag{3.33}$$

where $n = 0$ is excluded since some of the functions have a singular point at $t = 0$. Table 3.1 presents the results, where L_N and $\hat{L}_N$ are represented in their mantissa and exponent of 10. In the table, the positive number m shows the order of $F_j(s)$:

$$F_j(s) = O\left(\frac{1}{s^m} \right) \quad s \to \infty, \quad j = 1, \ldots, 17. \tag{3.34}$$

We can observe the correspondence between m and $L_N, \hat{L}_N$ in the table. That is, firstly, the error L_N is small for the large value of m, and secondly, the error is remarkably reduced by the proposed method especially for worse cases, $m \leq 1$. More specifically on the second point, there are improvements greater than 10^{-3} in magnitude of L_N's for those kinds of Laplace transforms, except for $F_{11}(s)$ and $F_{13}(s)$. The exceptions are the Laplace transforms which have discontinuities in the time domain. With the proposed method, the discontinuities are smoothened by the i-th difference and the errors look seemingly large at the discontinuous points. Figure 3.12 illustrates the inversions of $F_{11}(s)$ with four points around the discontinuous point $t = 5$ marked by the distinctive squares and crosses. Although the proposed method reduces the truncation error and avoids the Gibbs phenomenon, it yields several intermediate points due to the i-th difference.

Then, we compare the proposed method with the Durbin's technique of extending the window length N by K-times to reduce errors [9]. The window-extended inversion is given by

$$f_N^{(K)}(n\Delta t) = \frac{\mathrm{e}^{an\Delta t}}{T} \left\{ \mathrm{Re} \left[\sum_{k=0}^{N-1} \left(\sum_{\ell=0}^{K} F(a + \mathrm{j}(k + \ell N)\Delta\omega) \right) \mathrm{e}^{\mathrm{j}2\pi kn/N} \right] - \frac{1}{2} F(a) \right\} = f_{K \cdot N}(n\Delta t), \tag{3.35}$$

where the number of points in the time domain and the FFT length is unchanged to reduce the computational cost. In the last two columns of Table 3.1, the integers K_1, K_2 denote the minimum multipliers to achieve comparable small errors to the proposed method. Namely, defining the deviation of $f_N^{(K)}(t)$ by

$$L_N^{(K)} = \left[\frac{1}{N-1} \sum_{n=1}^{N-1} \left(f_N^{(K)}(n\Delta t) - f(n\Delta t) \right)^2 \right]^{1/2}, \tag{3.36}$$

we define K_1, K_2 as the minimum integers such that:

$$L_N^{(K_1)} < \hat{L}_N, \tag{3.37}$$

$$L_N^{(K_2)} < 10\hat{L}_N. \tag{3.38}$$

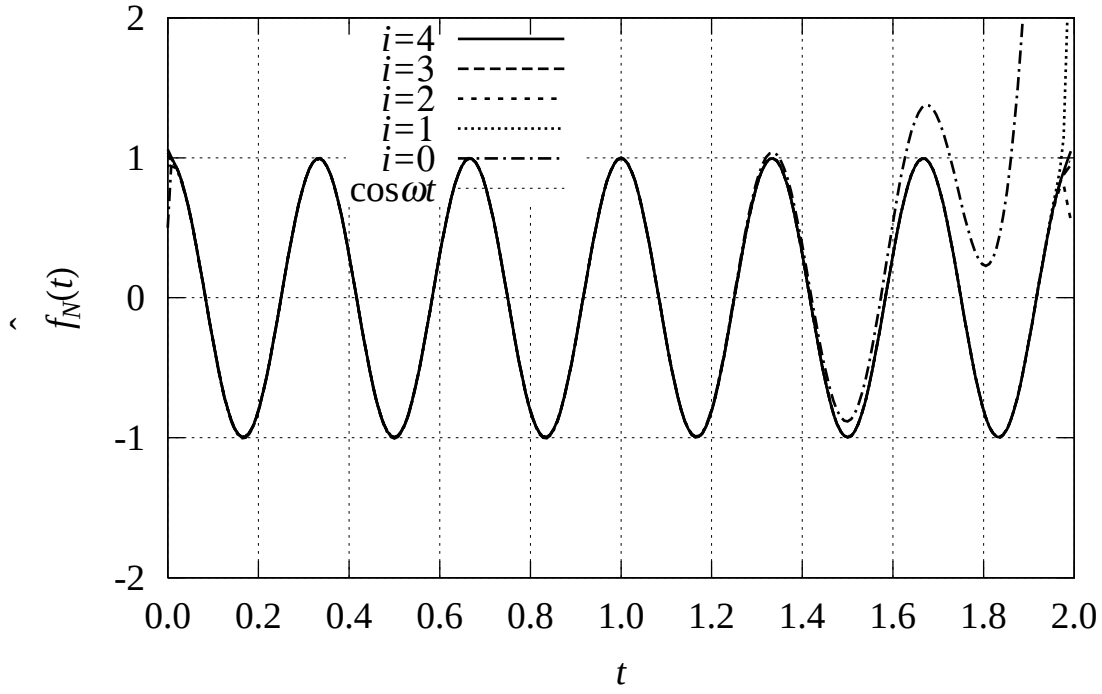


Figure 3.8: Proposed inversion of the cosine function $F(s) = s/(s^2 + \omega^2), \omega = 6\pi$ rad. ($N = 256, T = 1, a = 3.5$)

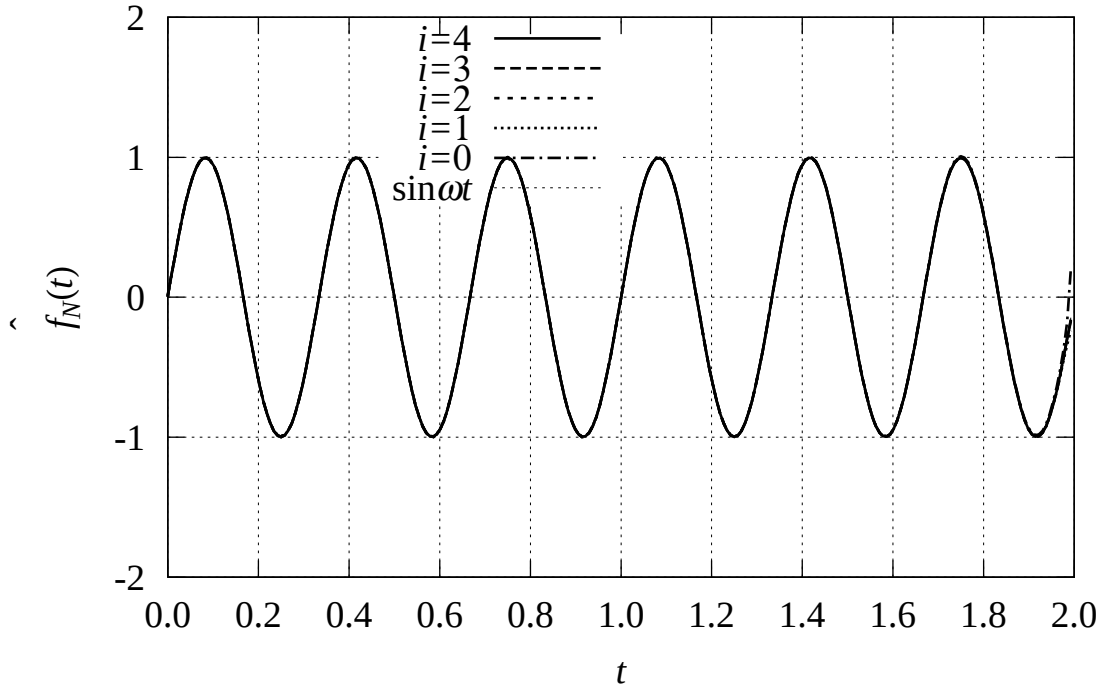


Figure 3.9: Proposed inversion of the sine function $F(s) = \omega/(s^2 + \omega^2)$, $\omega = 6\pi$ rad. ($N = 256, T = 1, a = 3.5$)

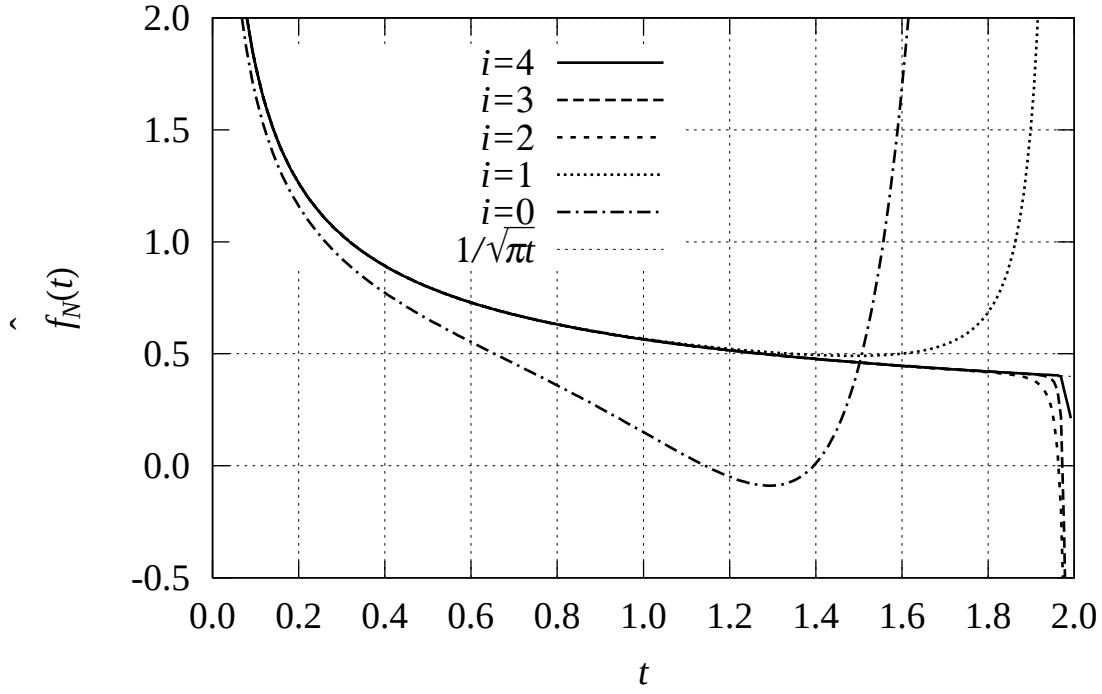


Figure 3.10: Proposed inversion of $F(s) = 1/\sqrt{s}$. ($N = 256, T = 1, a = 3.5$)

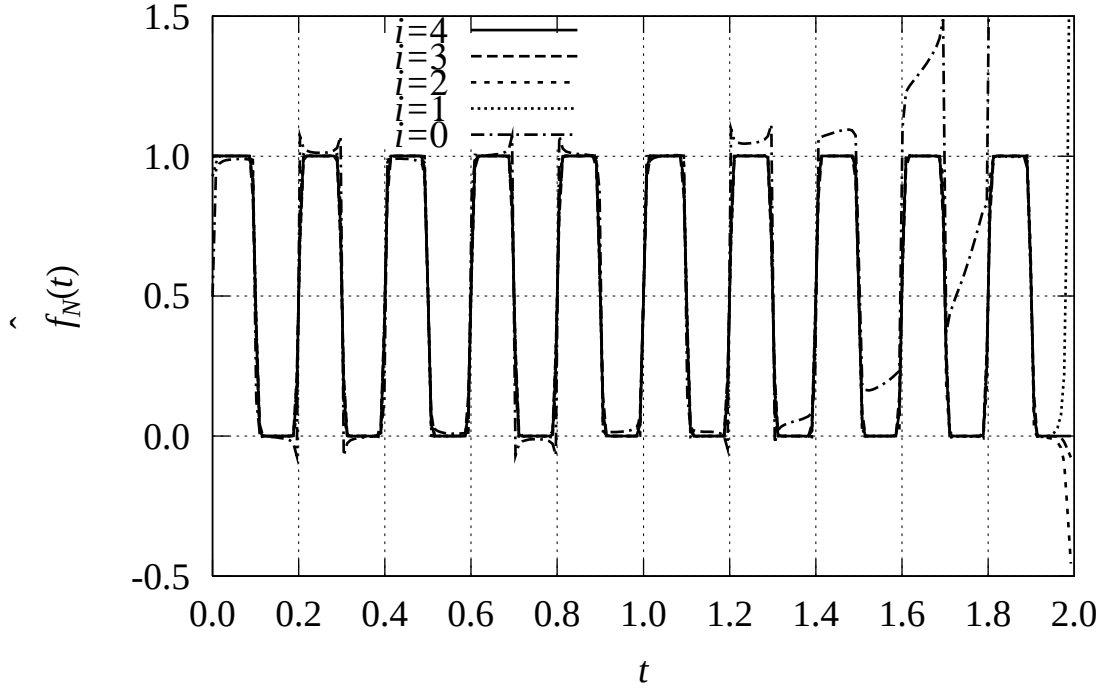


Figure 3.11: Proposed inversion of the square wave $F(s) = 1/s(1 + e^{-0.1s})$. ($N = 256, T = 1, a = 3.5$)

Table 3.1: Functions used in comparisons and the results ($N = 256, T = 15, aT = 3.5, i = 4$)

$F(s)$	$f(t)$	m	L_N	$\hat{L}_N$	K_1	K_2
$F_1(s) = \frac{1}{\sqrt{s^2+1}}$	$f_1(t) = J_0(t)$	1	4.03(±0)	6.17(−4)	–	608
$F_2(s) = \frac{1}{e^{1/s}\sqrt{s}}$	$f_2(t) = \frac{\cos(2\sqrt{t})}{\sqrt{\pi t}}$	1/2	1.90(+1)	1.51(−2)	–	–
$F_3(s) = \frac{1}{s+1/2}$	$f_3(t) = e^{-t/2}$	1	4.02(±0)	1.91(−4)	–	–
$F_4(s) = \frac{1}{(s+0.2)^2+1}$	$f_4(t) = e^{-0.2t}\sin t$	2	1.95(−2)	6.63(−4)	–	2
$F_5(s) = \frac{1}{s}$	$f_5(t) = 1$	1	4.03(±0)	9.25(−4)	–	425
$F_6(s) = \frac{1}{s^2}$	$f_6(t) = t$	2	4.99(−2)	4.18(−2)	–	1
$F_7(s) = \frac{1}{(s+1)^2}$	$f_7(t) = te^{-t}$	2	2.14(−2)	1.03(−3)	–	2
$F_8(s) = \frac{s}{s^2+1}$	$f_8(t) = \cos t$	1	4.03(±0)	1.83(−3)	1864	218
$F_9(s) = \frac{1}{s^2+1}$	$f_9(t) = \sin t$	2	1.91(−2)	2.99(−3)	–	1
$F_{10}(s) = \frac{1}{\sqrt{s}}$	$f_{10}(t) = \frac{1}{\sqrt{\pi t}}$	1/2	1.94(+1)	1.30(−2)	–	–
$F_{11}(s) = \frac{e^{-5s}}{s}$	$f_{11}(t) = \theta(t-5)$	1	8.16(−2)	1.98(−2)	3	1
$F_{12}(s) = \frac{\ln(s)}{s}$	$f_{12}(t) = -\gamma - \ln(t)$	–	1.53(+1)	1.44(−2)	–	–
$F_{13}(s) = \frac{1}{s(1+e^{-s})}$	$f_{13}(t)$ =square wave	1	5.31(±0)	1.24(−1)	–	5
$F_{14}(s) = \frac{s^2-1}{(s^2+1)^2}$	$f_{14}(t) = t\cos t$	2	3.24(−2)	4.10(−2)	1	1
$F_{15}(s) = \sqrt{s+1/2} - \sqrt{s+1/4}$	$f_{15}(t) = \frac{e^{-t/4} - e^{-t/2}}{\sqrt{4\pi t^3}}$	1/2	2.42(±0)	1.67(−3)	–	–
$F_{16}(s) = e^{-4\sqrt{s}}$	$f_{16}(t) = \frac{2e^{-4/t}}{\sqrt{\pi t^3}}$	–	3.47(−6)	3.53(−5)	1	1
$F_{17}(s) = \arctan\frac{1}{s}$	$f_{17}(t) = \frac{\sin t}{t}$	1	4.03(+1)	3.34(−4)	–	1061

In the columns L_N and $\hat{L}_N$, the representation $x(\pm n)$ means the value $x \times 10^{\pm n}$. In the columns K_1 and K_2, a dash means that the value is more than 100,000 at least.

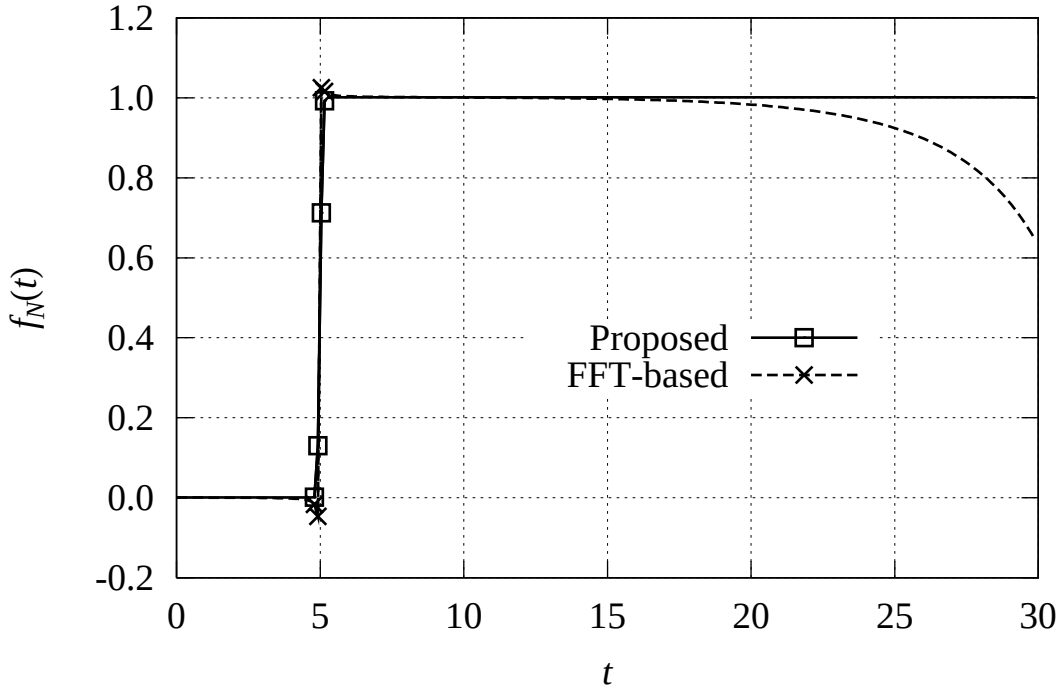


Figure 3.12: Inversion of the delayed unit step $F_{11}(s) = \mathrm{e}^{-5s}/s$, $f_{11}(t) = \theta(t-5)$. ($N = 256, T = 15, aT = 3.5, i = 4$)

In the two columns, a dash means that the condition was not satisfied for K less than 100,000. These results show that the proposed method is significantly effective to accelerate the convergence, and that the computation time required to attain the same accuracy with Durbin's technique is also much reduced.

3.5 Concluding Remarks

In this chapter, we analyzed errors introduced by the FFT-based method, and showed that the truncation error is the most part of the large errors observed in Section 2.3 and that the amount of the truncation error depends on the asymptotic behavior of the Laplace transform to be inverted. Then, we proposed a simple yet effective acceleration method in which the i-th power of $1/s$ suppresses the high-frequency components of $F(s)$ in the s domain and the i-th differentiation cancels $1/s^i$ in the time domain. We also analyzed errors introduced by the proposed method for $F(s) = 1/s$. Numerical examples showed that the proposed method reduces the truncation errors remarkably and extends the valid region to almost the entire $[0, 2T)$.

Chapter 4

Novel Numerical Laplace Transform Pair

4.1 Introduction

A pair of numerical Laplace transforms and their numerical inversion enables an efficient computation of the output waveform of linear time invariant systems given an arbitrary input waveform. As for transmission lines, it enables efficient analysis of frequency-dependent linear transmission lines terminated by non-linear loads [18]. The non-linear boundary condition has to be dealt with in the time domain while dealing with the transmission lines in the time domain is more time-consuming and sometimes less accurate. In fact, the skin effect term which is proportional to $1/\sqrt{s}$ is error-prone to invert numerically. With a numerical Laplace transform pair, one can solve the non-linear boundary conditions in the time domain, and handle the transmission lines in the complex frequency domain by linking the two domains.

The conventional Laplace transform pair described in Section 2.5 uses two kinds of interpolations to compensate for the halved number of data points due to the large truncation error and the aliasing effect. Our concern, when the transformations are repeatedly applied in turn, is that the interpolations might cause some errors, and especially that the interpolation of $f(t)$ may round discontinuities so that the function looks smoothly rising/falling around where it was originally discontinuous. To our surprise, however, numerical examples in the following section show that the repetition of the transformations does not yield such errors. We analyse the interpolations and clarify the reason why the interpolations work well. The analysis suggests that the interpolations are even unnecessary. We accordingly propose a novel numerical Laplace transform pair without interpolations, and further we apply the acceleration method proposed in Chapter 3 to reduce errors and extend the valid region.

4.2 Numerical Examples of Conventional Transform Pair

Figures 4.1 and 4.2 show the results of repeated applications of the conventional transform pair to the unit step. In each figure, the bottom figures are enlarged views of the top figures. In Fig. 4.1, the function is given at the beginning as $f_{\rm I}(t) = 1\,(t \geq 0)$ in the time domain, and in Fig.4.2 as $F_{\rm III}(s) = 1/s$ in the s domain, respectively. In both figures, the value of $f_{\rm I}(0)$ becomes its half after each repetition of the transform pair as indicated by Eq. (2.3). The second point $f_{\rm I}(\Delta\tau)$ is merely the linear interpolation of $f_{\rm I}(0)$ and $f_{\rm I}(2\Delta\tau)$ calculated by Eq. (2.36). In Fig. 4.2, the rising edge at $t = 0$ is slightly rounded since the function is given in the form of $F_{\rm III}(s) = 1/s$ with its high frequency components truncated at $k = N$ where $s = a + {\rm j}k\Delta\omega$. The errors introduced by the iteration are sufficiently small for both results and thus all the lines almost overlap each other except for $t = 0, \Delta\tau$. In fact, after 10 iterations, $|f_{\rm I}(t) - 1| < 0.011$ in Fig. 4.1 except for $t = 0, \Delta\tau$.

Then, Figs. 4.3 and 4.4 show the results for a delayed unit step given at the beginning as

$$f_{\rm I}(t) = \begin{cases} 0 & (t < 1/2) \\ 1/2 & (t = 1/2) \\ 1 & (t > 1/2) \end{cases} \tag{4.1}$$

and as $F_{\rm III}(s) = {\rm e}^{-s/2}/s$, respectively. The left and right points of the discontinuous point $t = 0.5$ are the linear interpolations of their neighboring points. The errors introduced by the repetition are again unexpectedly small even though the latter half of the transformed data is rejected and the former half is interpolated after each inverse/forward transformation.

4.3 Analysis of Interpolations

4.3.1 Interpolation of $F_{\rm II}(s)$

We investigate here what effect the interpolation of $F_{\rm II}(a + {\rm j}k\Delta\nu)$ into $F_{\rm III}(a + {\rm j}k\Delta\omega)$ brings about in the time domain, by starting with $f_{\rm I}(n\Delta\tau)$ and comparing it with $f_{\rm IV}(n\Delta t)$ obtained through the procedure I $\rightarrow$ II $\rightarrow$ III $\rightarrow$ IV in Fig. 2.7. Taking into account Eq. (2.30), we can combine the expressions for the interpolation Eqs. (2.31) and (2.32) into

$$F_{\rm III}(a + {\rm j}k\Delta\omega) = \frac{\pi}{\Omega}\left[\sum_{n=0}^{N-1} f_{\rm I}(n\Delta\tau){\rm e}^{-an\Delta\tau}{\rm e}^{-{\rm j}\pi nk/N} - \frac{1}{2}f_{\rm I}(0)\right], \quad k = 0, 1, \cdots, N-1. \tag{4.2}$$

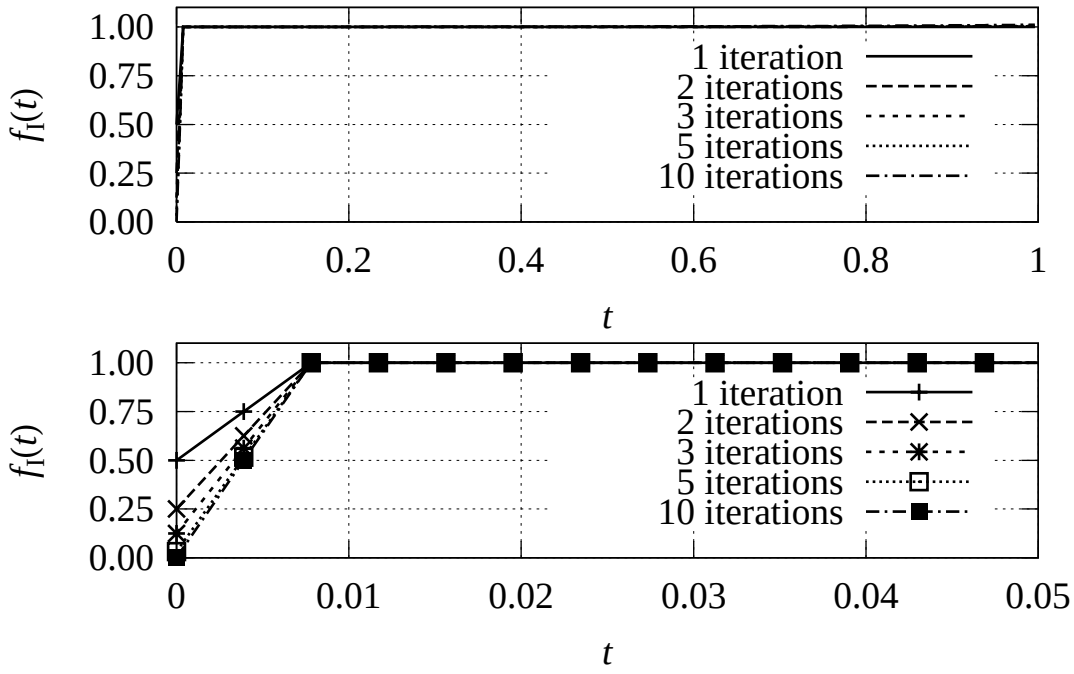


Figure 4.1: Results of repeated applications of the conventional transform pair to the unit step $f_{\mathrm{I}}(t) = 1$. ($N = 256, T = 1, a = 3.5$. The bottom figure is the enlarged view of the top figure.)

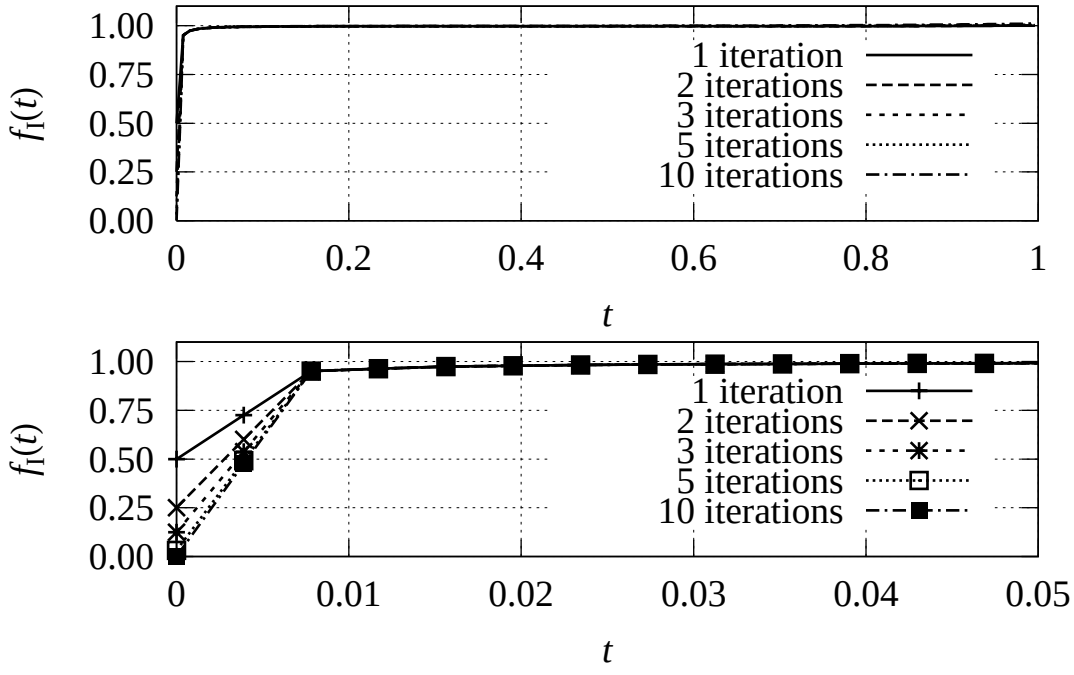


Figure 4.2: Results of repeated applications of the conventional transform pair to the unit step $F_{\mathrm{III}}(s) = 1/s$. ($N = 256, T = 1, a = 3.5$. The bottom figure is the enlarged view of the top figure.)

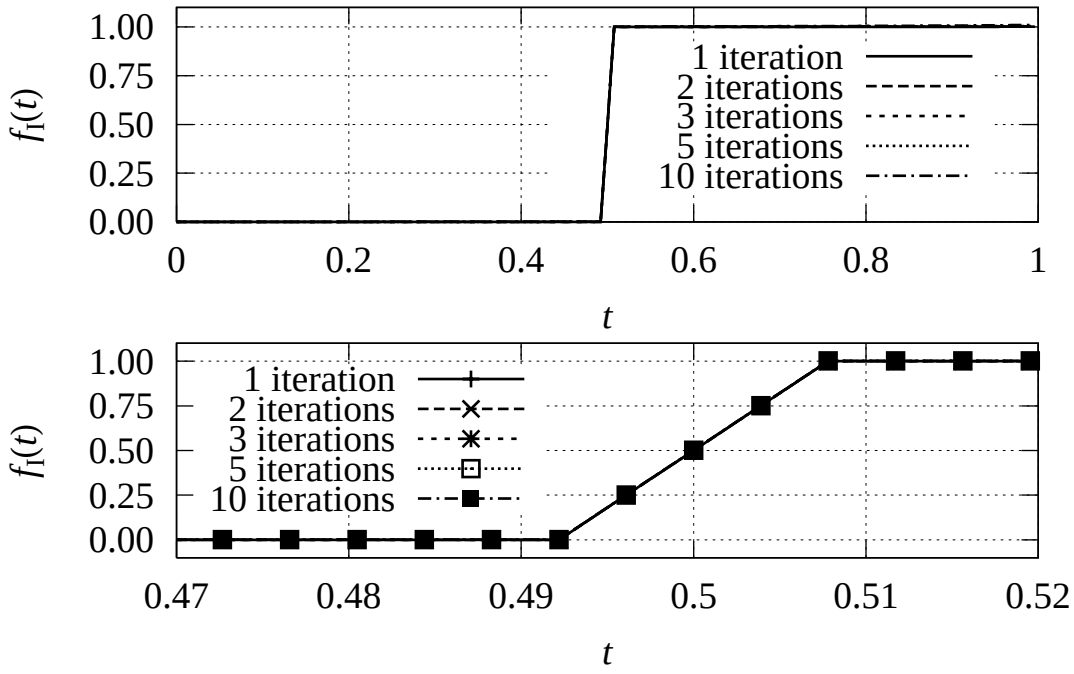


Figure 4.3: Results of repeated applications of the conventional transform pair to the delayed unit step $f_{\mathrm{I}}(t) = \theta(t - 0.5)$. ($N = 256, T = 1, a = 3.5$. The bottom figure is the enlarged view of the top figure.)

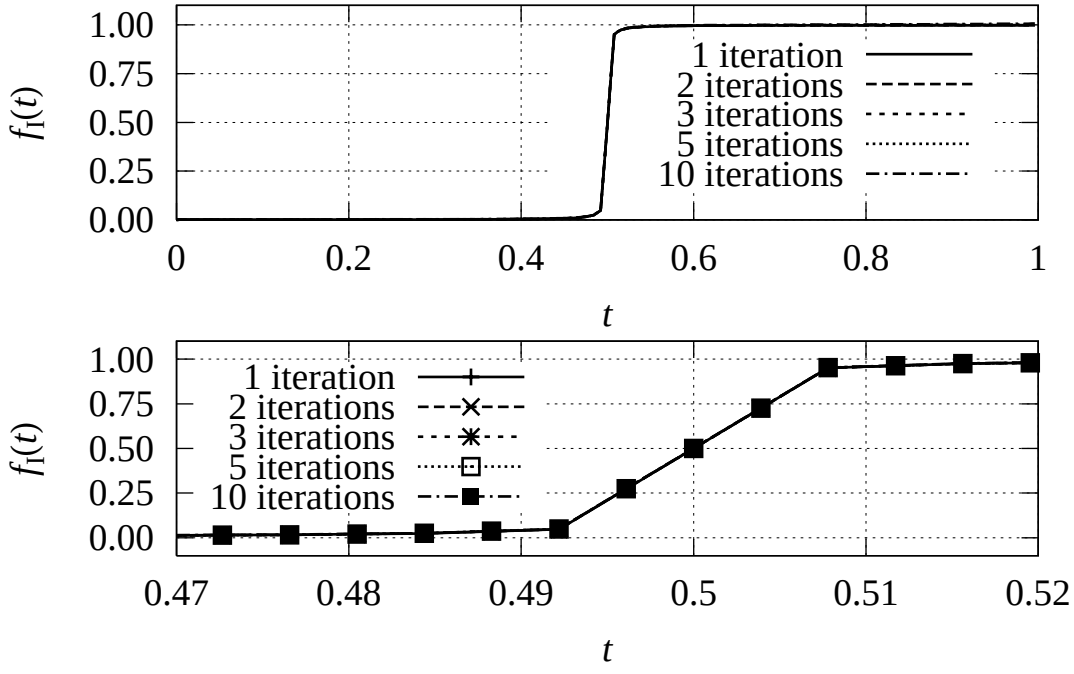


Figure 4.4: Results of repeated applications of the conventional transform pair to the delayed unit step $F_{\mathrm{III}}(s) = \mathrm{e}^{-s/2}/s$. ($N = 256, T = 1, a = 3.5$. The bottom figure is the enlarged view of the top figure.)

Then, the summation in Eq. (2.33) leads to

$$\sum_{k=0}^{N-1} F_{\mathrm{III}}(a+\mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}2\pi kn/N} = \frac{\pi}{\Omega}\sum_{k=0}^{N-1}\left[\sum_{m=0}^{N-1} f_{\mathrm{I}}(m\Delta\tau)\mathrm{e}^{-am\Delta\tau}\mathrm{e}^{-\mathrm{j}\pi mk/N} - \frac{1}{2}f_{\mathrm{I}}(0)\right]\mathrm{e}^{\mathrm{j}2\pi kn/N}$$
$$= \frac{\pi}{\Omega}\left[\sum_{m=0}^{N-1} f_{\mathrm{I}}(m\Delta\tau)\mathrm{e}^{-am\Delta\tau}\sum_{k=0}^{N-1}\mathrm{e}^{\mathrm{j}\pi k(2n-m)/N} - \frac{N\delta_n}{2}f_{\mathrm{I}}(0)\right] \tag{4.3}$$

where $\delta_n = 1$ if $n = 0$ and otherwise $\delta_n = 0$. For the whole Eq. (2.33), we have

$$f_{\mathrm{IV}}(n\Delta t) = \frac{\mathrm{e}^{an\Delta t}}{T}\frac{\pi}{\Omega}\left(\sum_{m=0}^{N-1} f_{\mathrm{I}}(m\Delta\tau)\mathrm{e}^{-am\Delta\tau}\mathrm{Re}\left[\sum_{k=0}^{N-1}\mathrm{e}^{\mathrm{j}\pi k(2n-m)/N}\right] - \frac{N\delta_n}{2}f_{\mathrm{I}}(0)\right.$$
$$\left. -\frac{1}{2}\left[\sum_{m=0}^{N-1} f_{\mathrm{I}}(m\Delta\tau)\mathrm{e}^{-am\Delta\tau} - \frac{1}{2}f_{\mathrm{I}}(0)\right]\right)$$
$$= \frac{\mathrm{e}^{an\Delta t}}{N}\left(\sum_{m=0}^{N-1} f_{\mathrm{I}}(m\Delta\tau)\mathrm{e}^{-am\Delta\tau}\left(\mathrm{Re}\left[\sum_{k=0}^{N-1}\mathrm{e}^{\mathrm{j}\pi k(2n-m)/N}\right] - \frac{1}{2}\right) - \frac{1}{2}\left(N\delta_n - \frac{1}{2}\right)f_{\mathrm{I}}(0)\right). \tag{4.4}$$

The term in the second bracket is the Dirichlet kernel. If $2n - m \equiv 0 \pmod{2N}$, the summation in the bracket is equal to N, and otherwise

$$\mathrm{Re}\left[\sum_{k=0}^{N-1}\mathrm{e}^{\mathrm{j}\pi k(2n-m)/N}\right] = \mathrm{Re}\left[\frac{1-\mathrm{e}^{\mathrm{j}\pi(2n-m)}}{1-\mathrm{e}^{\mathrm{j}\pi(2n-m)/N}}\right] = \frac{1}{2}\left(1-(-1)^{2n-m}\right) = \frac{1}{2}\left(1-(-1)^m\right). \tag{4.5}$$

Since $n, m = 0, 1, \cdots, N-1$, we finally have

$$f_{\mathrm{IV}}(n\Delta t) = \begin{cases} f_{\mathrm{I}}(2n\Delta\tau) & (0 \le n < N/2) \\ 0 & (N/2 \le n < N) \end{cases} - \frac{\delta_n}{2}f_{\mathrm{I}}(0) - \frac{\mathrm{e}^{an\Delta t}}{2N}\left[\sum_{m=0}^{N-1} f_{\mathrm{I}}(m\Delta\tau)\mathrm{e}^{-am\Delta\tau}(-1)^m - \frac{1}{2}f_{\mathrm{I}}(0)\right].$$
$$= \begin{cases} f_{\mathrm{I}}(2n\Delta\tau) & (0 \le n < N/2) \\ 0 & (N/2 \le n < N) \end{cases} - \frac{\delta_n}{2}f_{\mathrm{I}}(0) - \frac{\mathrm{e}^{an\Delta t}}{2T}F_{\mathrm{II}}(a+\mathrm{j}\Omega). \tag{4.6}$$

The first term shows that $f_{\mathrm{I}}(t)$ is extrapolated with zeros, and the fact is a consequence of the interpolation of $F_{\mathrm{II}}(s)$. The second term halves the value of $f_{\mathrm{I}}(0)$ as seen from Figs. 4.1 and 4.2. The third one is the error term derived by virtue of Eq. (2.30). Although it is potentially quite large due to the exponential, it remains small at least for $0 \le n < N/2$ if $F_{\mathrm{II}}(s)$ is a rapidly decreasing function in modulus.

Figure 4.5 illustrates the procedure taken in the above analysis on the left side, and its counterpart in the time domain on the right side. It should be noted that the extrapolated zeros are rejected in the subsequent step IV.

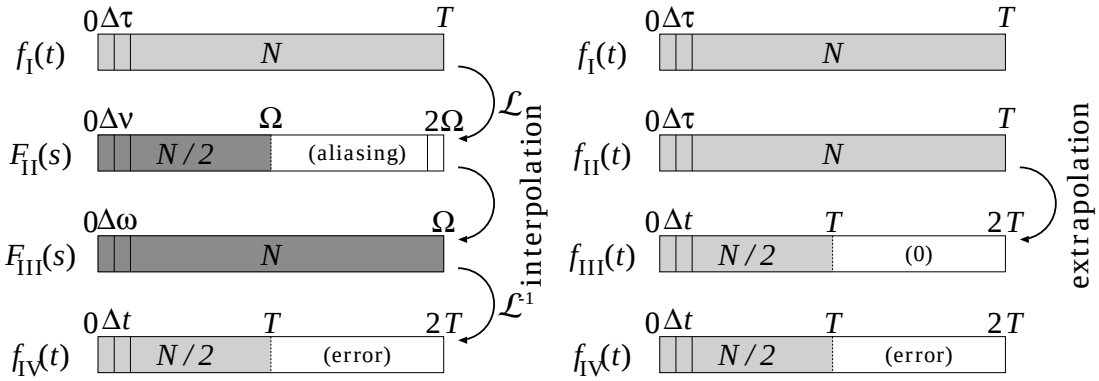


Figure 4.5: Interpolation of $F_{\mathrm{II}}(s)$.

4.3.2 Interpolation of $f_{\mathrm{IV}}(t)$

In this section, we analyse what effect the interpolation of $f_{\mathrm{IV}}(n\Delta t)$ into $f_{\mathrm{I}}(n\Delta\tau)$ brings about in the s domain, by starting with $F_{\mathrm{III}}(a+\mathrm{j}k\Delta\omega)$ and comparing it with $F_{\mathrm{II}}(a+\mathrm{j}k\Delta\nu)$ obtained through the procedure III $\to$ IV $\to$ I $\to$ II in Fig. 2.7. Since it is a little complicated than that of $F_{\mathrm{II}}(s)$, let us begin with the case with no interpolation of $f_{\mathrm{IV}}(t)$.

With no interpolation of $f_{\mathrm{IV}}(t)$ This case is equivalent to interpolating $f_{\mathrm{IV}}(t)$ with zeros as

$$f_{\mathrm{I}}(n\Delta\tau) = \begin{cases} f_{\mathrm{IV}}((n/2)\Delta t) & (n = 0, 2, \cdots, N-2) \\ 0 & (n = 1, 3, \cdots, N-1) \end{cases} . \tag{4.7}$$

Using the above equation in Eq. (2.30), we have

$$F_{\mathrm{II}}^{(\mathrm{N})}(a + \mathrm{j}k\Delta\nu) = \frac{\pi}{\Omega}\left[\sum_{n=0}^{N/2-1} f_{\mathrm{IV}}(n\Delta t)\mathrm{e}^{-an\Delta t}\mathrm{e}^{-\mathrm{j}4\pi kn/N} - \frac{1}{2}f_{\mathrm{IV}}(0)\right], \tag{4.8}$$

where superscript '(N)' denotes no interpolation. Before substituting Eq. (2.33) into the above, defining $F_{\mathrm{III}}(a + \mathrm{j}k\Delta\omega)$ for negative k as follows facilitates the calculation:

$$F_{\mathrm{III}}(a - \mathrm{j}k\Delta\omega) = F_{\mathrm{III}}^{*}(a + \mathrm{j}k\Delta\omega), \quad k = 0, 1, \cdots, N-1. \tag{4.9}$$

Then, we can rewrite Eq. (2.33) as

$$f_{\mathrm{IV}}(n\Delta t) = \frac{\mathrm{e}^{an\Delta t}}{2T}\sum_{k=-N+1}^{N-1} F_{\mathrm{III}}(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}2\pi kn/N}, \tag{4.10}$$

and substituting it into Eq. (4.8), we have

$$F_{\mathrm{II}}^{(\mathrm{N})}(a+\mathrm{j}k\Delta\nu) = \frac{\pi}{\Omega}\frac{1}{2T}\left[\sum_{n=0}^{N/2-1}\sum_{\ell=-N+1}^{N-1} F_{\mathrm{III}}(a+\mathrm{j}\ell\Delta\omega)\mathrm{e}^{\mathrm{j}2\pi\ell n/N}\mathrm{e}^{-\mathrm{j}4\pi kn/N} - \frac{1}{2}\sum_{\ell=-N+1}^{N-1} F_{\mathrm{III}}(a+\mathrm{j}\ell\Delta\omega)\right]$$
$$= \frac{1}{2N}\sum_{\ell=-N+1}^{N-1} F_{\mathrm{III}}(a+\mathrm{j}\ell\Delta\omega)\left[\sum_{n=0}^{N/2-1}\mathrm{e}^{\mathrm{j}2\pi(\ell-2k)n/N} - \frac{1}{2}\right]. \tag{4.11}$$

If $\ell - 2k \equiv 0 \pmod N$, the second summation is equal to $N/2$, and otherwise

$$\sum_{n=0}^{N/2-1}\mathrm{e}^{\mathrm{j}2\pi(\ell-2k)n/N} = \frac{1-\mathrm{e}^{\mathrm{j}\pi(\ell-2k)}}{1-\mathrm{e}^{\mathrm{j}2\pi(\ell-2k)/N}} = \frac{1-(-1)^{\ell}}{2}\left(1+\mathrm{j}\cot\frac{\pi(\ell-2k)}{N}\right)$$
$$= \begin{cases} 0 & (\ell:\text{even}) \\ 1+\mathrm{j}\cot\left(\dfrac{\pi(\ell-2k)}{N}\right) & (\ell:\text{odd}) \end{cases} \tag{4.12}$$

Hence, with mod N for congruences, we have

$$F_{\mathrm{II}}^{(\mathrm{N})}(a+\mathrm{j}k\Delta\nu) = \frac{1}{2N}\sum_{\ell=-N+1}^{N-1} F_{\mathrm{III}}(a+\mathrm{j}\ell\Delta\omega)\times\begin{cases} N/2-1/2 & (\ell-2k\equiv 0) \\ -1/2 & (\ell-2k\not\equiv 0, \ell:\text{even}) \\ 1/2+\mathrm{j}\cot(\pi(\ell-2k)/N) & (\ell-2k\not\equiv 0, \ell:\text{odd}) \end{cases}$$
$$= \frac{1}{4}\times\begin{cases} F_{\mathrm{III}}(a) & (k=0,N/2) \\ F_{\mathrm{III}}(a+\mathrm{j}2k\Delta\omega)+F_{\mathrm{III}}(a+\mathrm{j}(2k-N)\Delta\omega) & (0<k<N/2) \\ F_{\mathrm{III}}(a+\mathrm{j}(2k-N)\Delta\omega)+F_{\mathrm{III}}(a+\mathrm{j}(2k-2N)\Delta\omega) & (N/2<k<N) \end{cases}$$
$$+\frac{\mathrm{j}}{2N}\sum_{\substack{\ell=-N+1,\\ \ell:\text{odd}}}^{N-1} F_{\mathrm{III}}(a+\mathrm{j}\ell\Delta\omega)\cot\left(\frac{\pi(\ell-2k)}{N}\right) - \frac{1}{4N}\sum_{\ell=-N+1}^{N-1} F_{\mathrm{III}}(a+\mathrm{j}\ell\Delta\omega)(-1)^{\ell}$$
$$= \frac{1}{4}\times\begin{cases} F_{\mathrm{III}}(a) & (k=0,N/2) \\ F_{\mathrm{III}}(a+\mathrm{j}2k\Delta\omega)+F_{\mathrm{III}}^{*}(a+\mathrm{j}(N-2k)\Delta\omega) & (0<k<N/2) \\ F_{\mathrm{III}}(a+\mathrm{j}(2k-N)\Delta\omega)+F_{\mathrm{III}}^{*}(a+\mathrm{j}(2N-2k)\Delta\omega) & (N/2<k<N) \end{cases}$$
$$+\frac{\Delta\omega}{2\pi}\sum_{\substack{\ell=-N+1,\\ \ell:\text{odd}}}^{N-1} F_{\mathrm{III}}(a+\mathrm{j}\ell\Delta\omega)\cdot\frac{\pi}{\mathrm{j}\Omega}\cot\left(\frac{\pi(2k-\ell)}{N}\right) - \frac{T}{2N}f_{\mathrm{IV}}(T)\mathrm{e}^{-aT} \tag{4.13}$$

The first term includes $F_{\mathrm{III}}(a+\mathrm{j}k\Delta\omega)$ only for even k and is the aliased $F_{\mathrm{III}}(s)$ with the interval $N\Delta\omega = \Omega$. The second term in turn includes $F_{\mathrm{III}}(a+\mathrm{j}k\Delta\omega)$ only for odd k, and the cotangent factor is an aliased version of $1/(s-\sigma)$. Referring to Eq. (2.4), let $s = a+\mathrm{j}k\Delta\nu$ and $\sigma = a+\mathrm{j}\ell\Delta\omega$, then we have

$$\frac{1}{s-\sigma} = \frac{1}{\mathrm{j}k\Delta\nu - \mathrm{j}\ell\Delta\omega} = \frac{1}{\mathrm{j}(2k-\ell)\Delta\omega}. \tag{4.14}$$

In addition, a well known identity

$$\pi\cot(\pi x) = \sum_{i=-\infty}^{\infty} \frac{1}{x+i} \tag{4.15}$$

leads to

$$\frac{\pi}{\mathrm{j}\Omega}\cot\left(\frac{\pi(2k-\ell)}{N}\right) = \sum_{i=-\infty}^{\infty} \frac{1}{\mathrm{j}(2k-\ell)\Delta\omega + \mathrm{j}\Omega i}. \tag{4.16}$$

These equations evidence that the cotangent factor is the aliased $1/(s-\sigma)$ with the interval Ω. Hence, the first and second terms can be thought as the discretization of Eq. (2.4) with the aliasing of interval Ω. The third term is the error term derived by using Eq. (4.10).

In Figs. 4.8 and 4.9, the solid line in the third figure shows $F_{\mathrm{II}}^{(\mathrm{N})}(s)$ for an exponentially decaying sine function and the unit step function, respectively. We can see that $F_{\mathrm{II}}^{(\mathrm{N})}(s)$ is the aliased $F_{\mathrm{III}}(s)$ with interval Ω.

Interpolation by using $F_{\mathrm{III}}(s)$ We next consider the case where $f_{\mathrm{IV}}(t)$ is interpolated by using $F_{\mathrm{III}}(s)$. Taking into account Eq. (2.33), we can combine the expressions for the interpolation Eqs. (2.34) and (2.35) into

$$\begin{aligned} f_{\mathrm{I}}(n\Delta\tau) &= \frac{\mathrm{e}^{an\Delta\tau}}{T}\left\{\mathrm{Re}\left[\sum_{k=0}^{N-1} F_{\mathrm{III}}(a+\mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}\pi nk/N}\right] - \frac{1}{2}F_{\mathrm{III}}(a)\right\} \\ &= \frac{\mathrm{e}^{an\Delta\tau}}{2T}\sum_{k=-N+1}^{N-1} F_{\mathrm{III}}(a+\mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}\pi nk/N}, \quad n=0,1,\cdots,N-1. \end{aligned} \tag{4.17}$$

Using the above equation in Eq. (2.30), similar calculation as in the previous case gives

$$F_{\mathrm{II}}^{(\mathrm{F})}(a+\mathrm{j}k\Delta\nu) = \frac{1}{2N}\sum_{\ell=-N+1}^{N-1} F_{\mathrm{III}}(a+\mathrm{j}\ell\Delta\omega)\left[\sum_{n=0}^{N-1}\mathrm{e}^{\mathrm{j}\pi(\ell-2k)n/N} - \frac{1}{2}\right], \tag{4.18}$$

$$\sum_{n=0}^{N-1}\mathrm{e}^{\mathrm{j}\pi(\ell-2k)n/N} - \frac{1}{2} = \begin{cases} N-1/2 & (\ell-2k \equiv 0 \pmod{2N}) \\ -1/2 & (\ell-2k \not\equiv 0 \pmod{2N}, \ell : \text{even}) \\ 1/2 + \mathrm{j}\cot(\pi(\ell-2k)/2N) & (\ell-2k \not\equiv 0 \pmod{2N}, \ell : \text{odd}) \end{cases}, \tag{4.19}$$

$$\begin{aligned} F_{\mathrm{II}}^{(\mathrm{F})}(a+\mathrm{j}k\Delta\nu) = \frac{1}{2}\times &\begin{cases} F_{\mathrm{III}}(a+\mathrm{j}2k\Delta\omega) & (0 \le k < N/2) \\ 0 & (k = N/2) \\ F_{\mathrm{III}}^{*}(a+\mathrm{j}(2N-2k)\Delta\omega) & (N/2 < k < N) \end{cases} \\ &+ 2\cdot\frac{\Delta\omega}{2\pi}\sum_{\substack{\ell=-N+1, \\ \ell : \text{odd}}}^{N-1} F_{\mathrm{III}}(a+\mathrm{j}\ell\Delta\omega)\cdot\frac{\pi}{\mathrm{j}2\Omega}\cot\left(\frac{\pi(2k-\ell)}{2N}\right) - \frac{T}{2N}f_{\mathrm{IV}}(T)\mathrm{e}^{-aT}, \end{aligned} \tag{4.20}$$

where the superscript '(F)' denotes the interpolation by using $F_{\text{III}}(s)$. The first term includes $F_{\text{III}}(a + \mathrm{j}k\Delta\omega)$ only for even k and is the aliased $F_{\text{III}}(s)$ with the interval $2N\Delta\omega = 2\Omega$. The second term includes $F_{\text{III}}(a + \mathrm{j}k\Delta\omega)$ only for odd k, and the cotangent factor is the aliased $1/(s - \sigma)$ with interval 2Ω shown as

$$\frac{\pi}{\mathrm{j}2\Omega} \cot\left(\frac{\pi(2k - \ell)}{2N}\right) = \sum_{i=-\infty}^{\infty} \frac{1}{\mathrm{j}(2k - \ell)\Delta\omega + \mathrm{j}2\Omega i}. \tag{4.21}$$

Hence, we consider that the first and second terms form the discretization of Eq. (2.4) with the aliasing of interval 2Ω. The third term is the error term same as in the previous case.

The obtained $F_{\text{II}}^{(\text{F})}(s)$ satisfies the following relation:

$$F_{\text{II}}^{(\text{F})}(a + \mathrm{j}(N - k)\Delta\nu) = F_{\text{II}}^{(\text{F})^*}(a + \mathrm{j}k\Delta\nu), \quad k = 0, 1, \cdots, N - 1 \tag{4.22}$$

which shows that the latter half of $F_{\text{II}}^{(\text{F})}(s)$ is the reversed complex conjugate of the former half. We can see that $F_{\text{II}}^{(\text{F})}(s)$ is the aliased $F_{\text{III}}(s)$ with the interval 2Ω in the bottom figure of Figs. 4.8 and 4.9.

Linear interpolation Finally, we consider the case of the linear interpolation. In principle, convolution of $f_{\text{IV}}(n\Delta t)$ and a filter $h(t)$ presented by Fig. 4.6 yields the linear interpolation $f_{\text{I}}(n\Delta\tau)$ defined by Eq. (2.36) except for end points. The impulse response $h(t)$ and its Laplace transform $H(s)$ are given by

$$h(t) = \delta(t) + \frac{1}{2}(\delta(t + \Delta\tau) + \delta(t - \Delta\tau)), \tag{4.23}$$

$$H(s) = \mathcal{L}[h(t)] = 1 + \frac{1}{2}\left(\mathrm{e}^{-s\Delta\tau} + \mathrm{e}^{s\Delta\tau}\right) = 1 + \cosh(s\Delta\tau), \tag{4.24}$$

where we extend the integration range of Eq. (2.1) to $(-\infty, \infty)$ in order to include the term $\delta(t + \Delta\tau)$. Substituting $s = a + \mathrm{j}k\Delta\nu$, $H(s)$ becomes

$$H(a + \mathrm{j}k\Delta\nu) = 1 + \cosh\left(a\Delta\tau + \mathrm{j}\frac{2\pi k}{N}\right) \tag{4.25}$$

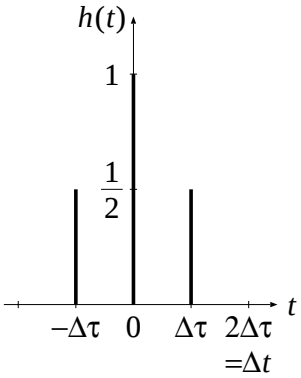


Figure 4.6: The linear interpolation filter $h(t)$.

and supposing $\Delta\tau = T/N \to 0$ by taking $N \to \infty$, we have

$$H(a + \mathrm{j}k\Delta\nu) = 1 + \cos\left(\frac{2\pi k}{N}\right). \tag{4.26}$$

Then, $H(a + \mathrm{j}k\Delta\nu)$ takes its maximum value 2 at the end points $k = 0, N$ and its minimum value 0 at the mid point $k = N/2$. The filter accordingly emphasizes low frequency components and suppresses high frequency components sinusoidally in the s domain.

Using $H(s)$, the linear interpolation of $f_{\mathrm{IV}}(t)$ gives

$$\begin{aligned} F_{\mathrm{II}}^{(\mathrm{L})}(a + \mathrm{j}k\Delta\nu) = {} & H(a + \mathrm{j}k\Delta\nu)F_{\mathrm{II}}^{(\mathrm{N})}(a + \mathrm{j}k\Delta\nu) \\ & - \frac{\pi}{2\Omega}\sinh\left(a\Delta\tau + \mathrm{j}\frac{2\pi k}{N}\right) f_{\mathrm{IV}}(0) \\ & + \frac{\pi}{2\Omega} f_{\mathrm{IV}}\left(\left(\frac{N}{2} - 1\right)\Delta t\right)\mathrm{e}^{-a(N-1)\Delta\tau}\mathrm{e}^{-\mathrm{j}2\pi k(N-1)/N} \end{aligned} \tag{4.27}$$

where the superscript '(L)' denotes the linear interpolation. The additional second term deals with $-f_{\mathrm{I}}(0)/2$ in Eq. (2.30) and subtracts a non-existent term introduced by the non-causal property of the filter $h(t)$. The third term is a treatment of irregularity of the end point defined by the second equation of Eq. (2.36). The obtained $F_{\mathrm{II}}^{(\mathrm{L})}(s)$ satisfies

$$F_{\mathrm{II}}^{(\mathrm{L})}(a + \mathrm{j}(N - k)\Delta\nu) = F_{\mathrm{II}}^{(\mathrm{L})^*}(a + \mathrm{j}k\Delta\nu), \quad k = 0, 1, \cdots, N - 1, \tag{4.28}$$

which shows that the latter half is the reversed complex conjugate of the former half. In Figs. 4.8 and 4.9, the dashed line in the third figure shows $F_{\mathrm{II}}^{(\mathrm{L})}(s)$.

Let us look at the principal term $H(s)F_{\mathrm{II}}^{(\mathrm{N})}(s)$ of Eq. (4.27). Although the aliasing interval of $F_{\mathrm{II}}^{(\mathrm{N})}$ is Ω, the linear interpolation filter $H(s)$ suppresses the frequency components around $\mathrm{Im}[s] = \Omega$, and thereby the aliasing interval of $F_{\mathrm{II}}^{(\mathrm{L})}$ becomes like 2Ω as shown in Figs. 4.8 and 4.9.

Summary We briefly summarize the interpolation of $f_{\mathrm{IV}}(t)$ and compare $F_{\mathrm{II}}^{(\mathrm{F})}(s)$ and $F_{\mathrm{II}}^{(\mathrm{L})}(s)$ with each other. Figure 4.7 illustrates the procedure taken in the above analyses on the left side, and its counterpart in the s domain on the right. The interpolation of $f_{\mathrm{IV}}(t)$, by either using $F_{\mathrm{III}}(s)$ or using the linear interpolation, extrapolates $F_{\mathrm{IV}}(s)$ with its reversed complex conjugate in a sense that Eqs. (4.22) and (4.28) hold. It must be noted that the extrapolated components are rejected in the subsequent step II.

We mention errors introduced by the interpolations. In Fig. 4.8, $F_{\mathrm{II}}^{(\mathrm{L})}(s)$ still has two small spurious peaks at left and right of $\mathrm{Im}[s] = \Omega$ originating from the aliasing of $F_{\mathrm{II}}^{(\mathrm{N})}(s)$ with the interval Ω, while $F_{\mathrm{II}}^{(\mathrm{F})}(s)$ has no such spurious peaks. In fact, $f_{\mathrm{I}}(t)$ contains larger errors when linearly interpolated than interpolated by using $F_{\mathrm{III}}(s)$. On the contrary, in Fig. 4.9, the single

spurious peak of $F_{\text{II}}^{(\text{L})}(s)$ is almost completely suppressed by the filter $H(s)$. This is because the unit step is a special kind of function that can be exactly restored by the linear interpolation. As we showed in Fig. 2.8, however, the interpolation by using $F_{\text{III}}(s)$ yields oscillatory errors of frequency $\Omega/2\pi = 1/2\Delta\tau$. There is a tiny spurious peak at $\text{Im}[s] = \Omega$ in $F_{\text{II}}^{(\text{F})}(s)$ correspondingly.

Hence, it depends on the frequency content that which of $F_{\text{II}}^{(\text{F})}(s)$ or $F_{\text{II}}^{(\text{L})}(s)$ is better than the other. Taking large N and using linear interpolation would be a safe decision. Regardless of the choice, however, some errors are introduced by the interpolation of $f_{\text{IV}}(t)$.

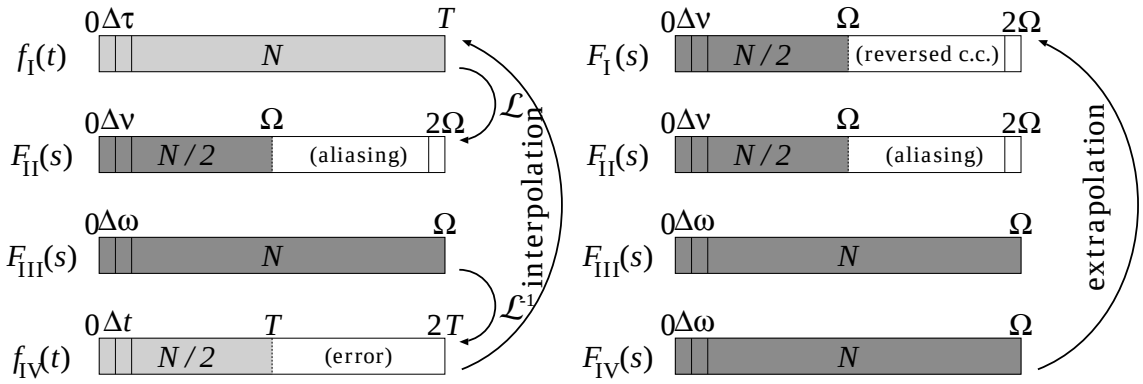


Figure 4.7: Interpolation of $f_{\text{IV}}(t)$.

4.4 Summary of Conventional Transform Pair

Let us summarize here the conventional numerical Laplace transform pair:

1. The latter half of $f_{\text{IV}}(n\Delta t)$ is discarded since it contains the large truncation error. The latter half of $F_{\text{II}}(a + \text{j}k\Delta\nu)$ is discarded since it is reversed complex conjugate of the former half, i.e. the aliasing.

2. The transform pair compensate for the halved number of valid data points by using interpolations. In spite of the use of interpolations, the errors introduced by the repeated applications of the transform pair are small even for discontinuous points.

3. The interpolation of $F_{\text{II}}(s)$ into $F_{\text{III}}(s)$ corresponds to the extrapolation of $f_{\text{IV}}(t)$ to $[T, 2T)$ with zeros. The extrapolated portion is to be discarded in the subsequent step IV.

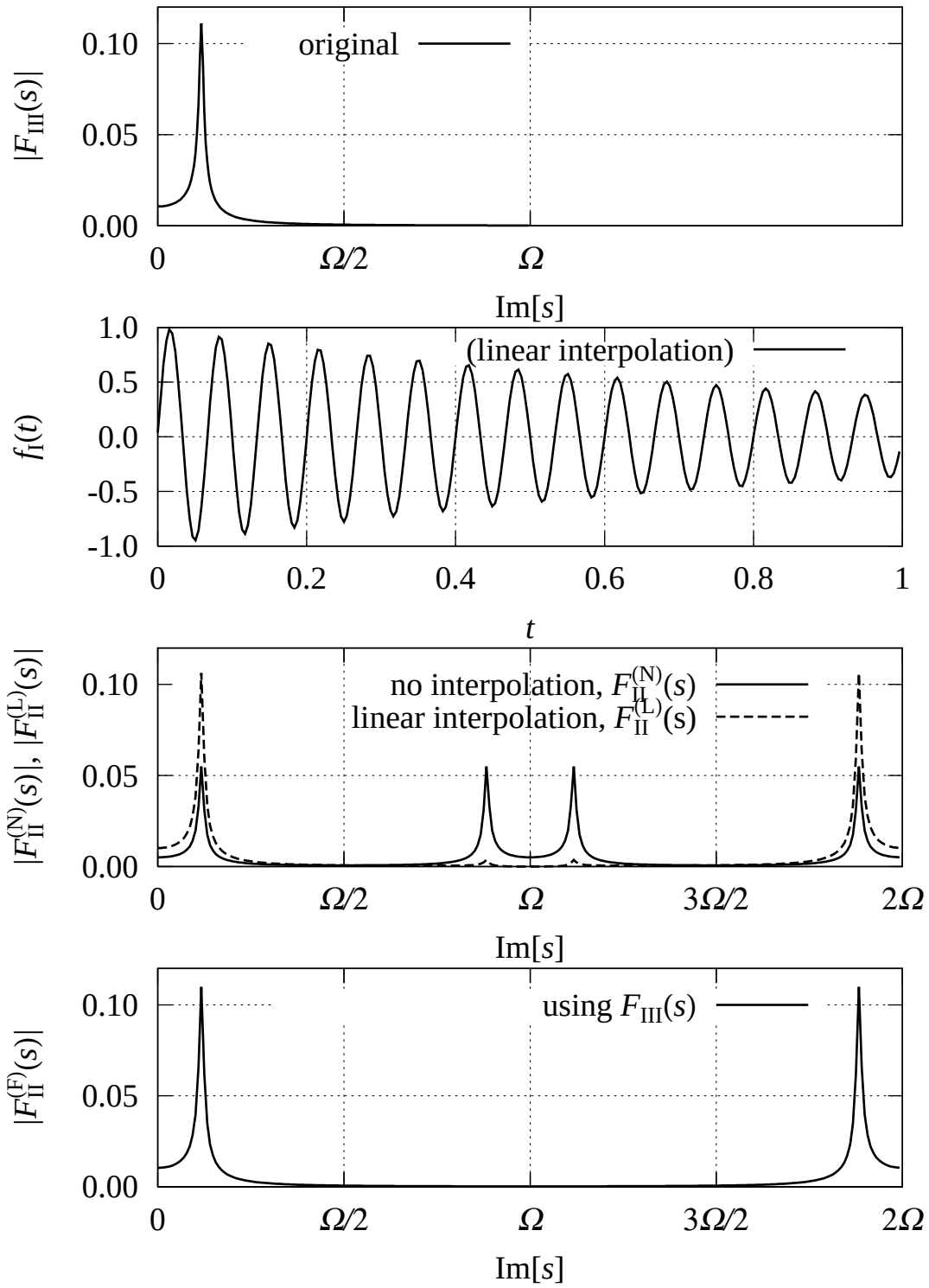


Figure 4.8: The types of interpolations of $f_{\mathrm{IV}}(t)$ and resulting $F_{\mathrm{II}}(s)$. ($F_{\mathrm{III}}(s) = \omega/((s+1)^2 + \omega^2), \omega = 30\pi, N = 256, T = 1, a = 3.5$)

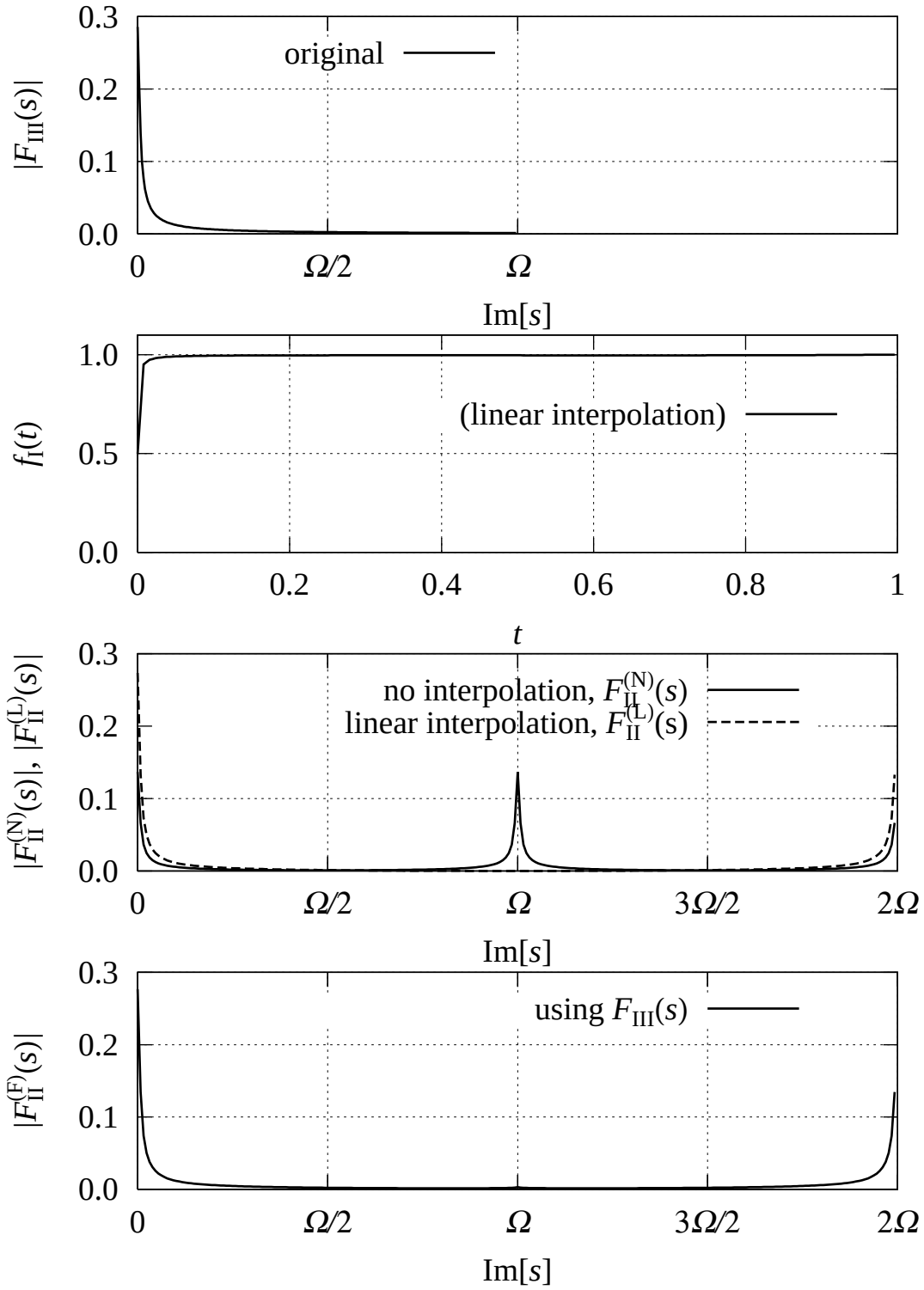


Figure 4.9: The types of interpolations of $f_{IV}(t)$ and resulting $F_{II}(s)$. ($F_{III}(s) = 1/s, N = 256, T = 1, a = 3.5$)

4. The interpolation of $f_{\mathrm{IV}}(t)$ into $f_{\mathrm{I}}(t)$ corresponds to the extrapolation of $F_{\mathrm{II}}(s)$ to $[\Omega, 2\Omega)$ with its reversed complex conjugate. The extrapolated portion is to be discarded in the subsequent step II.

5. The interpolations introduce errors as shown in Eqs. (4.6), (4.20) and (4.27). The errors are small but not zero, and thereby the transform pair is not numerically reversible like the discrete Fourer transform pair.

Considering the items 3 and 4 conversely, the discarded latter half in the steps II and IV turns out to be the extrapolated portion in the previous steps I and III, respectively, except for the very first time discarding. This fact is the reason why the repeated applications of the pair yields only small amount of errors. Namely, no substantial information is lost by the rejection of the latter half of transformed data, but is the made-up data points by the interpolations.

The above consideration might suggest that there is no use in extrapolating the latter half and discarding it. However, the rejection cancels out the large truncation errors caused by an inverse transformation and the aliasing effect by a forward transformation, and the interpolation compensates for the discarded data. Namely, the pair of the rejection and the interpolation is a technique to defeat the large errors and the aliasing effect. The rejection and interpolation technique, however, needs some computations and introduces some errors. We propose another simple technique in the following section.

4.5 Novel Numerical Laplace Transform Pair

4.5.1 Transform Pair without Interpolations

The conventional transform pair keeps N number of data points by using interpolations. However, it turned out that only $N/2$ number of data points is truly valid. We eliminate the interpolations by explicitly dealing with the $N/2$ number of valid data points, and at the same time, holding another $N/2$ number of data points to handle the truncation errors and the aliasing. That is, we develop a transform pair in which the latter half of $f(n\Delta t)$ is filled with zeros, and the latter half of $F(a + \mathrm{j}k\Delta\omega)$ is filled with its reversed complex conjugate as shown in Fig. 4.10.

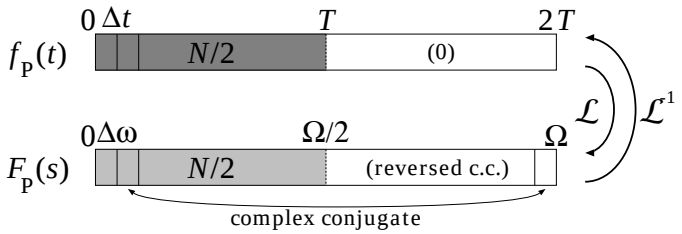


Figure 4.10: The transform pair without interpolations

We define the numerical inversion by

$$f_{\mathrm{P}}(n\Delta t) = \begin{cases} \dfrac{\mathrm{e}^{an\Delta t}}{2T} \displaystyle\sum_{k=0}^{N-1} F_{\mathrm{P}}(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}2\pi kn/N} & \left(0 \le n < \dfrac{N}{2}\right) \\ 0 & \left(\dfrac{N}{2} \le n < N\right) \end{cases}, \tag{4.29}$$

where the latter half of $F_{\mathrm{P}}(a + \mathrm{j}k\Delta\omega)$ is the reversed complex conjugate as

$$F_{\mathrm{P}}(a + \mathrm{j}k\Delta\omega) = F_{\mathrm{P}}^{*}(a + \mathrm{j}(N-k)\Delta\omega), \quad k = N/2+1, N/2+2, \cdots, N-1. \tag{4.30}$$

Then, we define the numerical Laplace transform of $f_{\mathrm{P}}(t)$ by

$$F_{\mathrm{P}}(a + \mathrm{j}k\Delta\omega) = \frac{2\pi}{\Omega}\left[\sum_{n=0}^{N/2-1} f_{\mathrm{P}}(n\Delta t)\mathrm{e}^{-an\Delta t}\mathrm{e}^{-\mathrm{j}2\pi kn/N} - \frac{1}{2}f_{\mathrm{P}}(0)\right], \quad 0 \le k < N. \tag{4.31}$$

So defined $F_{\mathrm{P}}(a + \mathrm{j}k\Delta\omega)$ satisfies Eq. (4.30). Note that the computational cost of Eq. (4.31) can be reduced by employing a modified FFT algorithm. Equation (4.29) can also be computed effectively by using the symmetry of Eq. (4.30).

Numerical Examples

Figures 4.11~4.14 show the results of repeated applications of the above proposed no-interpolation transform pair to the unit step and the delayed unit step. In Figs. 4.11 and 4.13, the function was given as

$$f_{\mathrm{P}}(n\Delta t) = \begin{cases} f(n\Delta t) & (0 \le n < N/2) \\ 0 & (N/2 \le n < N) \end{cases} \tag{4.32}$$

where $f(t) = 1$ and $f(t) = \theta(t - 0.5)$, respectively, and in Figs. 4.12 and 4.14, it was given as

$$F_{\mathrm{P}}(a + \mathrm{j}k\Delta\omega) = \begin{cases} F(a + \mathrm{j}k\Delta\omega) & (0 \le k < N/2) \\ F^{*}(a + \mathrm{j}(N-k)\Delta\omega) & (N/2 \le k < N) \end{cases} \tag{4.33}$$

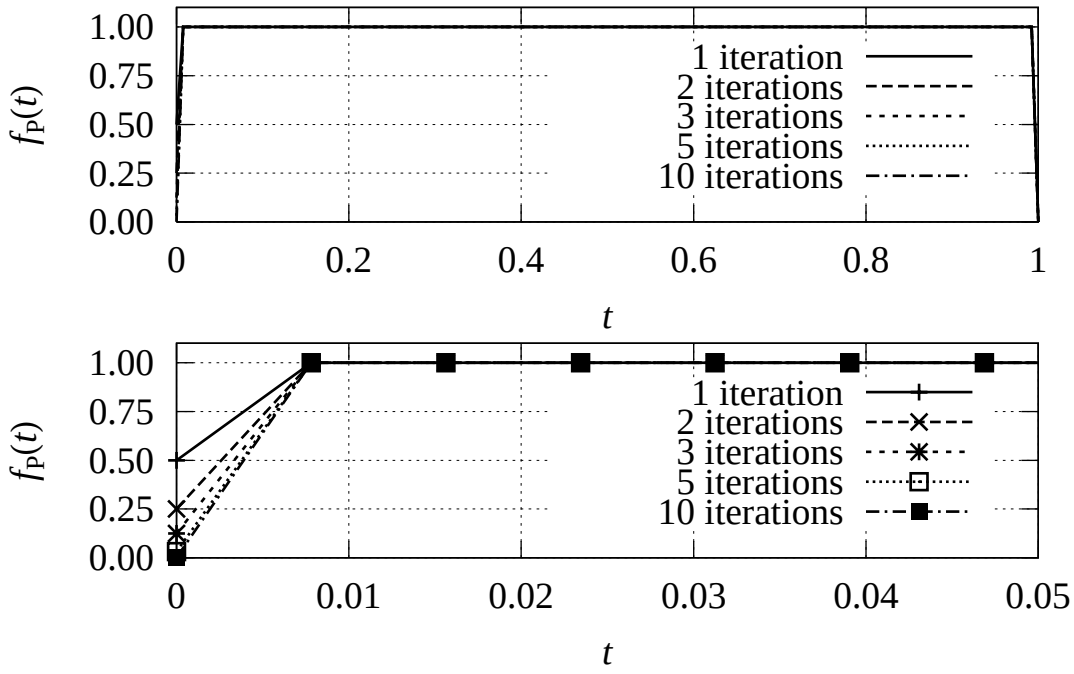


Figure 4.11: Results of repeated applications of the no-interpolation transform pair to the unit step $f(t) = 1$. ($N = 256, T = 1, a = 3.5$)

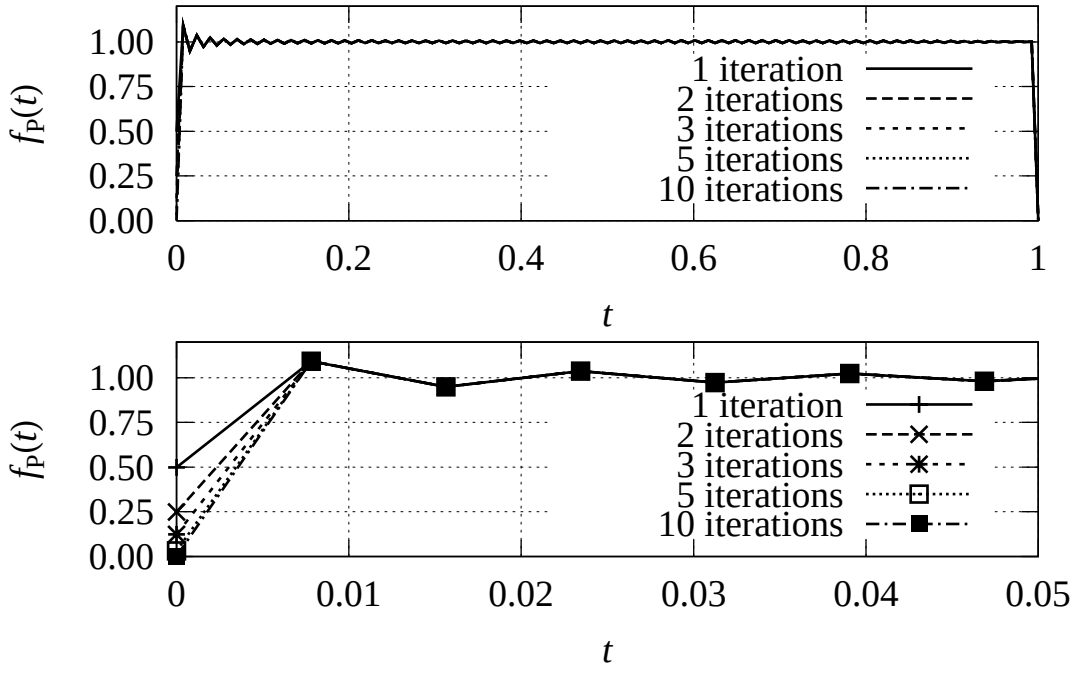


Figure 4.12: Results of repeated applications of the no-interpolation transform pair to the unit step $F(s) = 1/s$. ($N = 256, T = 1, a = 3.5$)

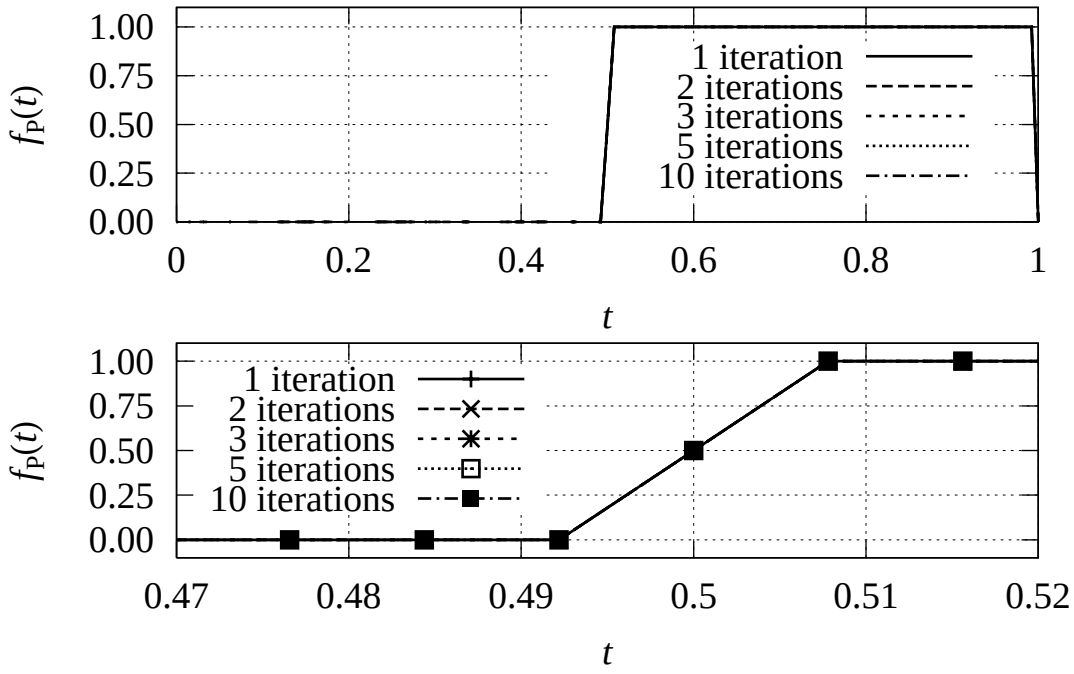


Figure 4.13: Results of repeated applications of the no-interpolation transform pair to the delayed unit step $f(t) = \theta(t - 0.5)$. ($N = 256, T = 1, a = 3.5$)

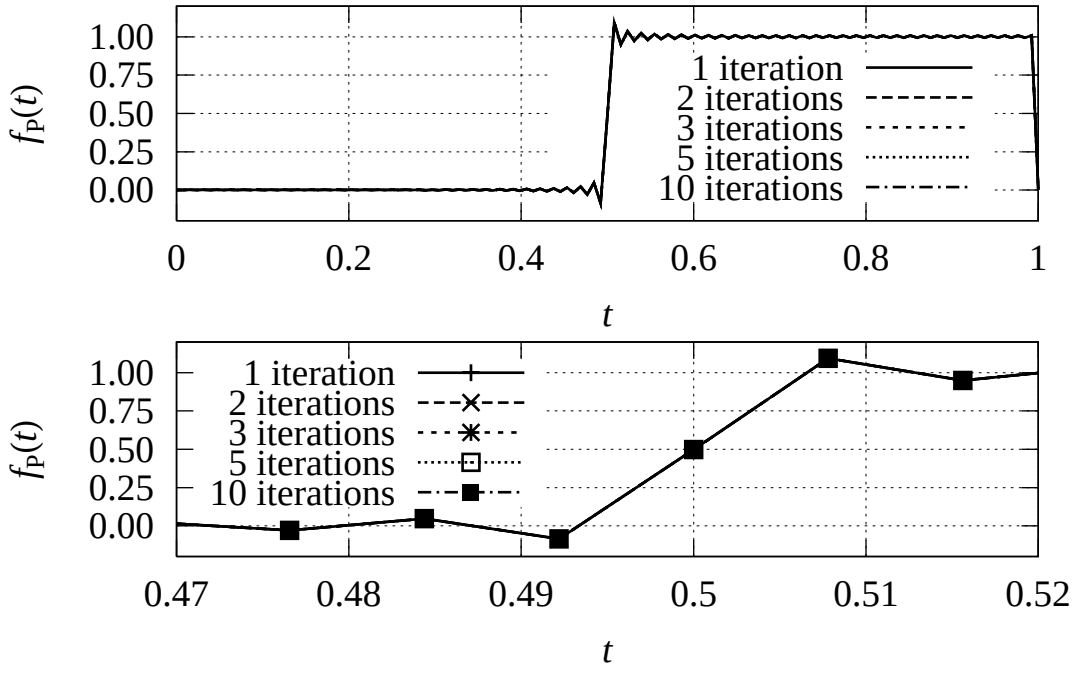


Figure 4.14: Results of repeated applications of the no-interpolation transform pair to the delayed unit step $F(s) = \mathrm{e}^{-s/2}/s$. ($N = 256, T = 1, a = 3.5$)

where $F(s) = 1/s$ and $F(s) = \mathrm{e}^{-s/2}/s$, respectively.

In Figs. 4.11 and 4.13, there is no error other than the halved $f_{\mathrm{P}}(0)$ through the repeated applications. In Figs. 4.12 and 4.14, the oscillatory error is caused by the first numerical inversion of $F_{\mathrm{P}}(s)$. As we pointed out in Section 2.5, the oscillation stems from the lack of bandwidth, namely one needs $F(s)$ of bandwidth $[0, \Omega)$ to obtain $f(t)$ of sampling period Δt. However, once $f_P(n\Delta t)$ is obtained by the first numerical inversion, there is no error as well other than the halved $f_{\mathrm{P}}(0)$ in the subsequent repeated applications of the no-interpolation transform pair.

Error Analysis

We analyse here the reason why the application of Eq. (4.31) and Eq. (4.29) in this order to $f_{\mathrm{P}}(n\Delta t)$ gives no error other than $f_{\mathrm{P}}(0)$. Suppose we have $g_{\mathrm{P}}(n\Delta t)$ by the application of Eqs. (4.31) and (4.29) to $f_{\mathrm{P}}(n\Delta t)$. Then it is given by

$$\begin{aligned} g_{\mathrm{P}}(n\Delta t) &= \frac{\mathrm{e}^{an\Delta t}}{2T}\frac{2\pi}{\Omega}\sum_{k=0}^{N-1}\left[\sum_{m=0}^{N/2-1} f_{\mathrm{P}}(m\Delta t)\mathrm{e}^{-am\Delta t}\mathrm{e}^{-\mathrm{j}2\pi km/N} - \frac{1}{2}f_{\mathrm{P}}(0)\right]\mathrm{e}^{\mathrm{j}2\pi kn/N} \\ &= \frac{1}{N}\left[\sum_{m=0}^{N/2-1} f_{\mathrm{P}}(m\Delta t)\mathrm{e}^{a(n-m)\Delta t}\sum_{k=0}^{N-1}\mathrm{e}^{\mathrm{j}2\pi k(n-m)/N} - \frac{N\delta_n}{2}f_{\mathrm{P}}(0)\right] \\ &= \begin{cases} f_{\mathrm{P}}(n\Delta t) & (0 \le n < N/2) \\ 0 & (N/2 \le n < N) \end{cases} - \frac{\delta_n}{2}f_{\mathrm{P}}(0). \end{aligned} \tag{4.34}$$

Thus, $g_{\mathrm{P}}(n\Delta t)$ exactly reproduces $f_{\mathrm{P}}(n\Delta t)$ except for $f_{\mathrm{P}}(0)$. The difference from Eq. (4.6), the case $F_{\mathrm{II}}(s)$ is interpolated by using $f_{\mathrm{I}}(t)$, is the absence of the error term. The error term stems from the rejection of the mid frequency component $F_{\mathrm{II}}(a + \mathrm{j}\Omega)$ in the course of the interpolation of $F_{\mathrm{II}}(s)$ into $F_{\mathrm{III}}(s)$. The no-interpolation pair, however, keeps the mid frequency component $F_{\mathrm{P}}(a + \mathrm{j}\Omega/2)$ through the transformations and thereby the error term disappears.

4.5.2 Use of The Proposed Numerical Inversion

In this section, we apply the accelerated numerical inversion proposed in Chapter 3 to $f_{\mathrm{P}}(t)$ in order to extend the valid region. Defining the denominator $D_i(a + \mathrm{j}k\Delta\omega)$ as

$$D_i(a + \mathrm{j}k\Delta\omega) = \begin{cases} (a + \mathrm{j}k\Delta\omega)^i & (0 \le k < N/2) \\ (a + \mathrm{j}(k-N)\Delta\omega)^i & (N/2 \le k < N) \end{cases}, \tag{4.35}$$

we define the i-th accelerated $f_{\mathrm{P}}(t)$ as

$$G_{\mathrm{P}}^{(i)}(a + \mathrm{j}k\Delta\omega) = \frac{F_{\mathrm{P}}(a + \mathrm{j}k\Delta\omega)}{D_i(a + \mathrm{j}k\Delta\omega)}, \tag{4.36}$$

$$g_{\mathrm{P}}^{(i)}(n\Delta t) = \frac{\mathrm{e}^{an\Delta t}}{2T}\sum_{k=0}^{N-1} G_{\mathrm{P}}^{(i)}(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}2\pi kn/N}, \tag{4.37}$$

$$f_{\mathrm{P}}^{(i)}(n\Delta t) = \frac{\Delta^i}{(\Delta t)^i} g_{\mathrm{P}}^{(i)}(n\Delta t), \tag{4.38}$$

where $n = 0, 1, \cdots, N-1$, and the difference operator Δ^i is defined as in Section 3.3.1. Figure 4.15 shows $f_{\mathrm{P}}^{(i)}(t)$ ($i = 0, 1, \cdots, 4$) for the unit step. Note that $f_{\mathrm{P}}^{(0)}(t)$ is equal to $f_{\mathrm{P}}(t)$ without the replacement of the latter half with zeros. Due to the lack of the bandwidth, the truncation error for $f_{\mathrm{P}}^{(0)}(t)$ is quite large and oscillatory for the latter half. As for $i \geq 1$, the error is markedly reduced when i is odd while it still remains large when i is even although we can see a minor improvement.

We consider that the dependence on the parity of i originates from an oscillatory property of $g_{\mathrm{P}}^{(i)}(n\Delta t)$ and the central difference, Eqs. (3.18) and (3.19). From Eqs. (4.30) and (4.35), we have

$$g_{\mathrm{P}}^{(i)}(n\Delta t) = \frac{\mathrm{e}^{an\Delta t}}{T}\left\{\mathrm{Re}\left[\sum_{k=0}^{N/2-1} G_{\mathrm{P}}^{(i)}(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}2\pi kn/N}\right] - \frac{1}{2}G_{\mathrm{P}}^{(i)}(a) + \frac{1}{2}G_{\mathrm{P}}^{(i)}(a + \mathrm{j}\Omega/2)\mathrm{e}^{\mathrm{j}\pi n}\right\}. \tag{4.39}$$

If n is even and given as $n = 2m$, $g_{\mathrm{P}}^{(i)}(n\Delta t)$ becomes

$$g_{\mathrm{P}}^{(i)}(m(2\Delta t)) = \frac{\mathrm{e}^{am(2\Delta t)}}{T}\left\{\mathrm{Re}\left[\sum_{k=0}^{N/2-1} G_{\mathrm{P}}^{(i)}(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}2\pi km/(N/2)}\right] - \frac{1}{2}G_{\mathrm{P}}^{(i)}(a) + \frac{1}{2}G_{\mathrm{P}}^{(i)}(a + \mathrm{j}\Omega/2)\right\}, \tag{4.40}$$

and thus $g_{\mathrm{P}}^{(i)}(m(2\Delta t))$ gives, as defined by Eq. (2.14), ordinary $N/2$-point numerical inversion of $G_{\mathrm{P}}^{(i)}(a + \mathrm{j}k\Delta\omega)$ of sampling period $2\Delta t$ with the additional monotonic third term. Hence, $g_{\mathrm{P}}^{(i)}(m(2\Delta t))$ does not contain the oscillation as a function of m. Similarly, if n is odd and given as $n = 2m + 1$, $g_{\mathrm{P}}^{(i)}(n\Delta t)$ becomes

$$g_{\mathrm{P}}^{(i)}(m(2\Delta t)+\Delta t) = \frac{\mathrm{e}^{am(2\Delta t)}}{T}\left\{\mathrm{Re}\left[\sum_{k=0}^{N/2-1} G_{\mathrm{P}}^{(i)}(a + \mathrm{j}k\Delta\omega)\mathrm{e}^{(a+\mathrm{j}k\Delta\omega)\Delta t}\mathrm{e}^{\mathrm{j}2\pi km/(N/2)}\right] - \frac{G_{\mathrm{P}}^{(i)}(a)}{2} - \frac{G_{\mathrm{P}}^{(i)}(a + \mathrm{j}\frac{\Omega}{2})}{2}\right\}, \tag{4.41}$$

and then $g_{\mathrm{P}}^{(i)}(m(2\Delta t) + \Delta t)$ gives ordinary $N/2$-point inversion of Δt-shifted $G_{\mathrm{P}}^{(i)}(a + \mathrm{j}k\Delta\omega)$ of sampling period $2\Delta t$ with the additional monotonic third term. Thus, $g_{\mathrm{P}}^{(i)}(m(2\Delta t) + \Delta t)$ has no such oscillation as a function of m either. On the other hand, the central difference for odd i defined by Eq. (3.19) involves only every other neighboring points while that for even i defined by Eq. (3.18) involves all adjoining points. Accordingly, if i is odd, the difference operation does

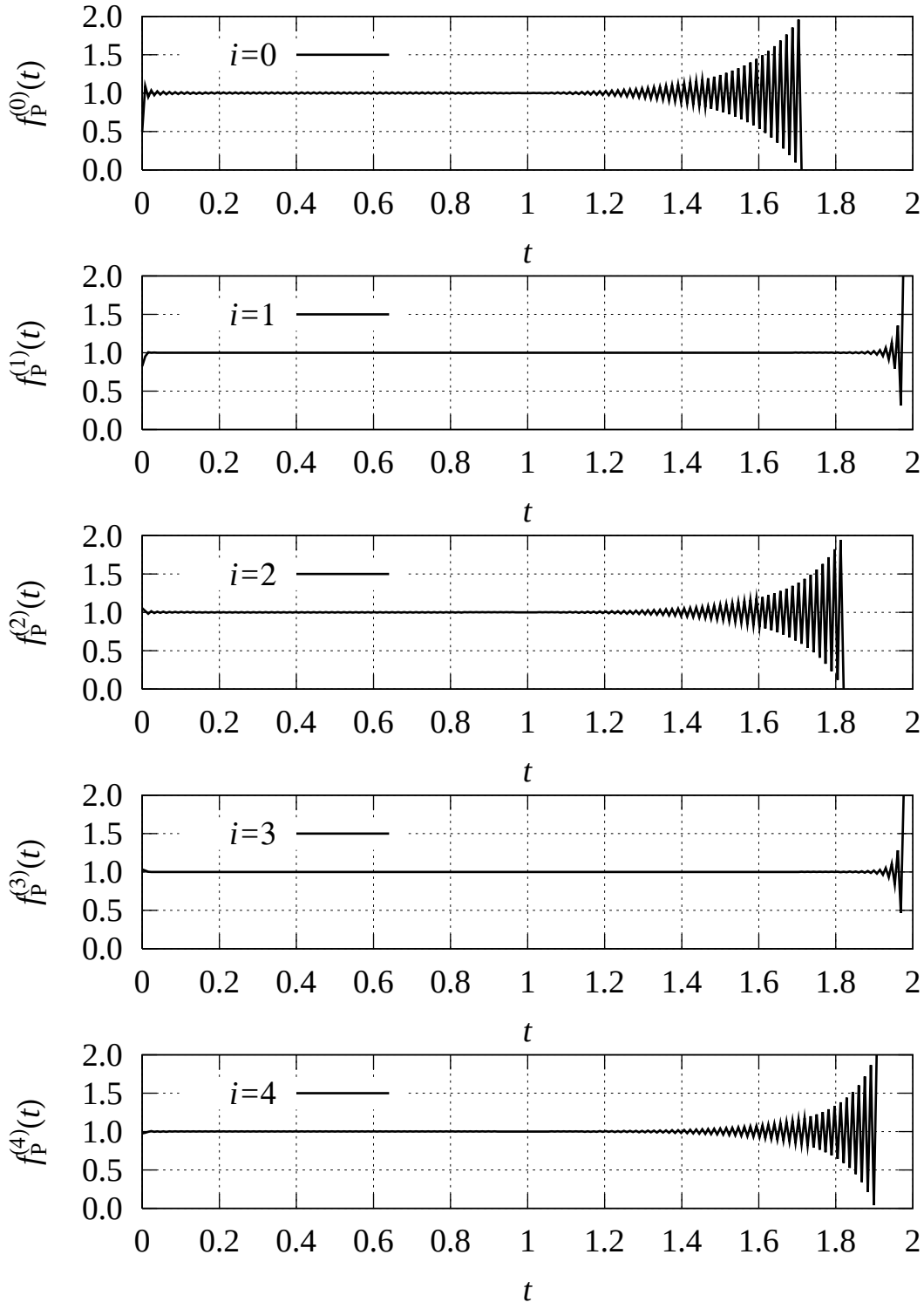


Figure 4.15: The accelerated $f_{Pi}(t)$ for the unit step $F(s) = 1/s$. ($N = 256, T = 1, a = 3.5$)

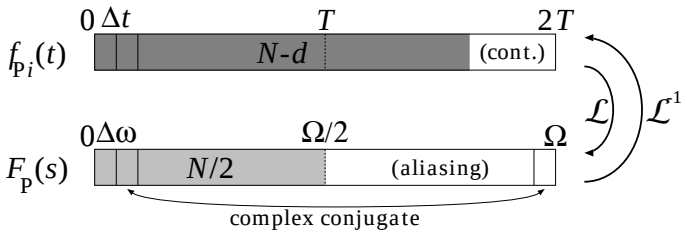


Figure 4.16: The transform pair with the proposed numerical inversion

not interfere with the oscillation in $g_P^{(i)}(n\Delta t)$ except for several end points. That is why $f_P^{(i)}(n\Delta t)$ gives better results if i is odd.

Now, we propose a novel transform pair using the proposed numerical inversion (Fig. 4.16):

$$f_{Pi}(n\Delta t) = \begin{cases} f_P^{(i)}(n\Delta t) & (0 \le n < N-d) \\ f_P^{(i)}((N-d-1)\Delta t) & (N-d \le n < N) \end{cases}, \tag{4.42}$$

where d is an integer specifying the rejection length. The last d points are filled with $f_P^{(i)}((N-d-1)\Delta t)$ instead of zeros in order not to generate a discontinuity which can cause errors due to the i-th difference. We use Eq. (4.31) for the forward transformation.

Figures 4.17~4.20 show the results of repeated applications of the proposed transform pair to the unit step and the delayed unit step with $N = 256, d = 32$. The last d points, namely $t \ge 1.75$, are nothing but merely the simple copies of $f_P^{(i)}((N-d-1)\Delta t)$. Hence, the valid region is extended by 1.75 times in these cases. In Figs. 4.17 and 4.18, although some errors are introduced around $t = 0$, $f_{Pi}(0)$ is no longer halved by using the proposed inversion. In Figs. 4.19 and 4.20, the rising edge is rounded with the repeated applications. This is a fundamental drawback of the proposed inversion due to the numerical differentiation, but can be remedied by taking large N.

4.6 Examples of Filtering

As an example of the numerical Laplace transform pairs, consider a filtering application as shown in Fig. 4.21. Here we first give $f(t) = e^{-2t}$, and choose

$$H(s) = \frac{s-2}{s+2} \tag{4.43}$$

as the filter function. The filtering loop generates the following Laguerre functions:

$$F_n(s) = \frac{(s-2)^n}{(s+2)^{n+1}}, \quad n = 0, 1, \cdots. \tag{4.44}$$

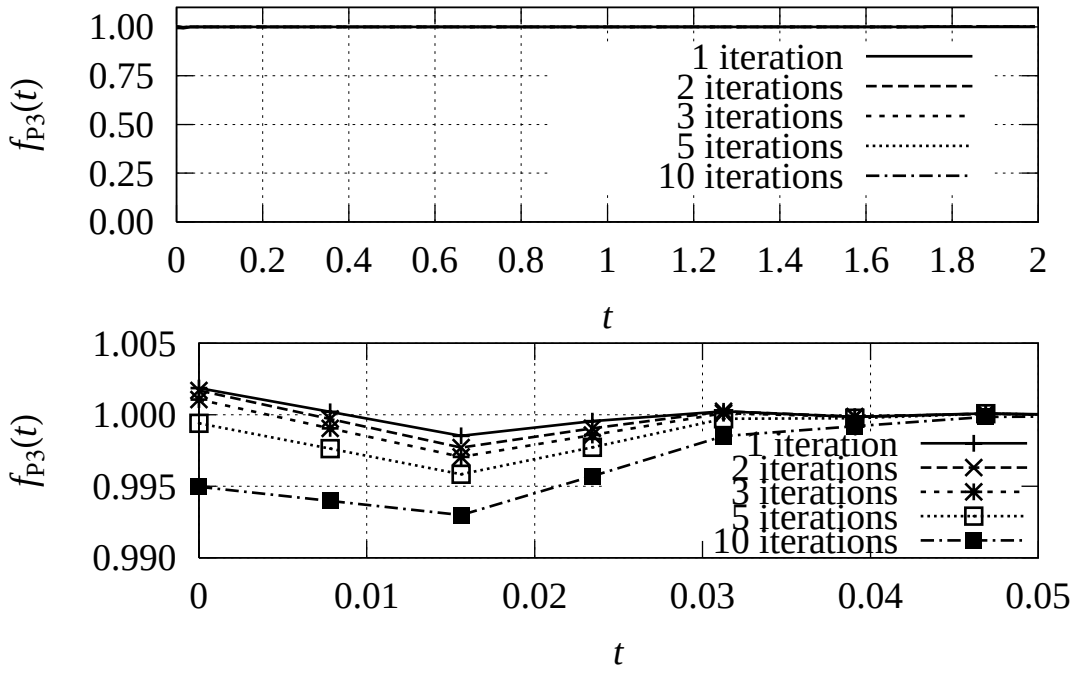


Figure 4.17: Results of repeated applications of the proposed transform pair to the unit step $f(t) = 1$. ($N = 256, T = 1, a = 3.5, i = 3, d = 32$)

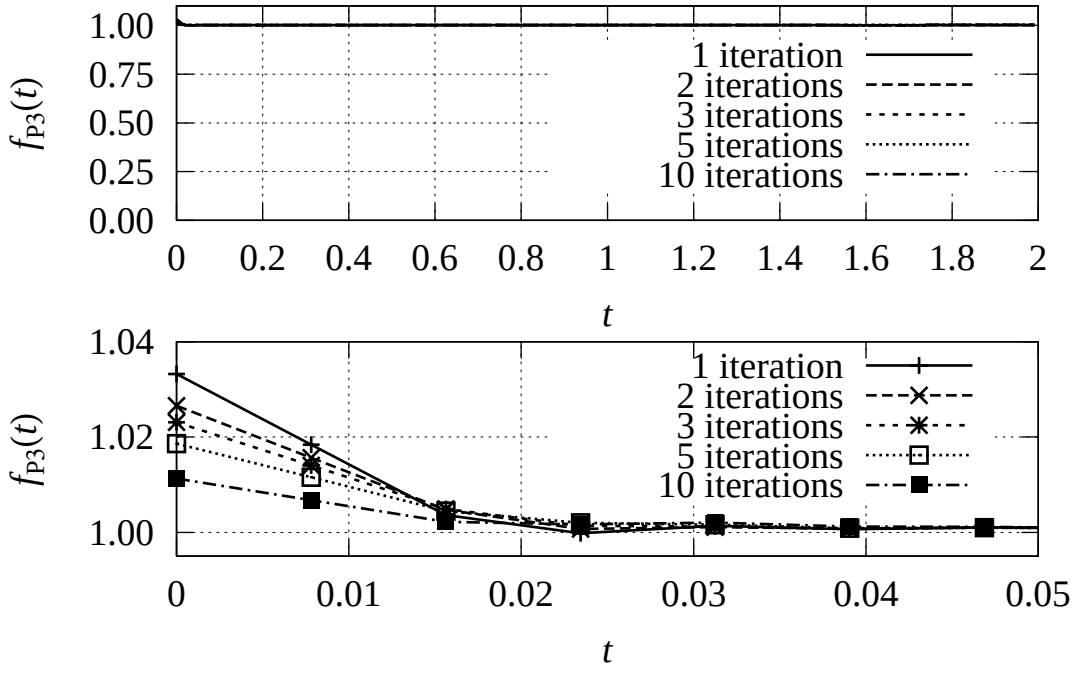


Figure 4.18: Results of repeated applications of the proposed transform pair to the unit step $F(s) = 1/s$. ($N = 256, T = 1, a = 3.5, i = 3, d = 32$)

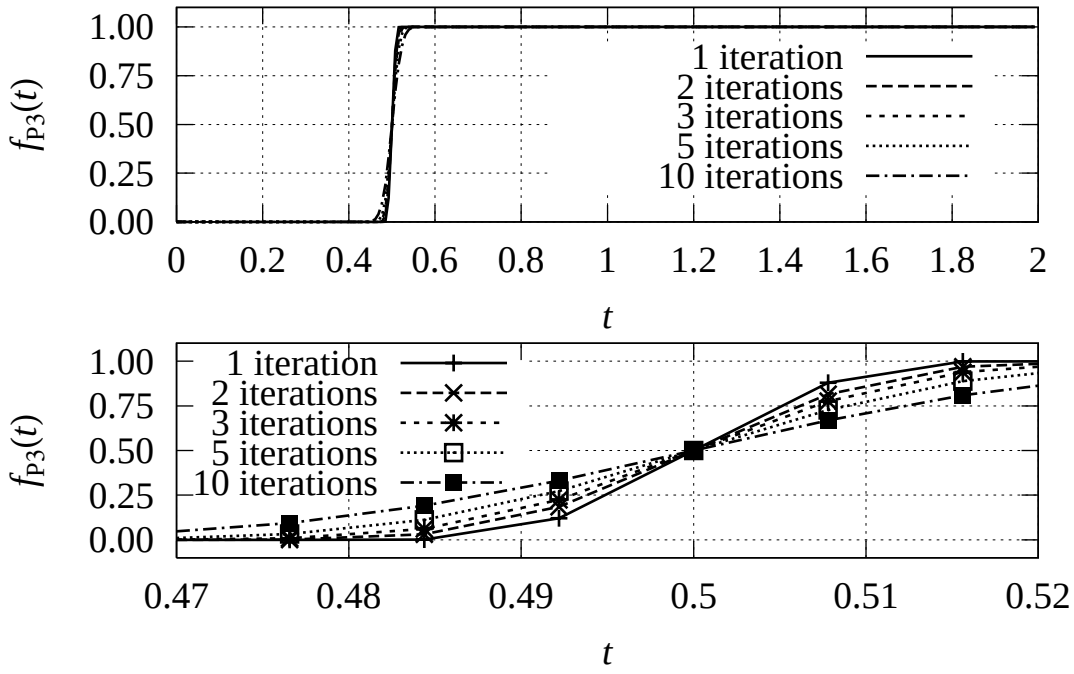


Figure 4.19: Results of repeated applications of the proposed transform pair to the delayed unit step $f(t) = \theta(t - 0.5)$. ($N = 256, T = 1, a = 3.5, i = 3, d = 32$)

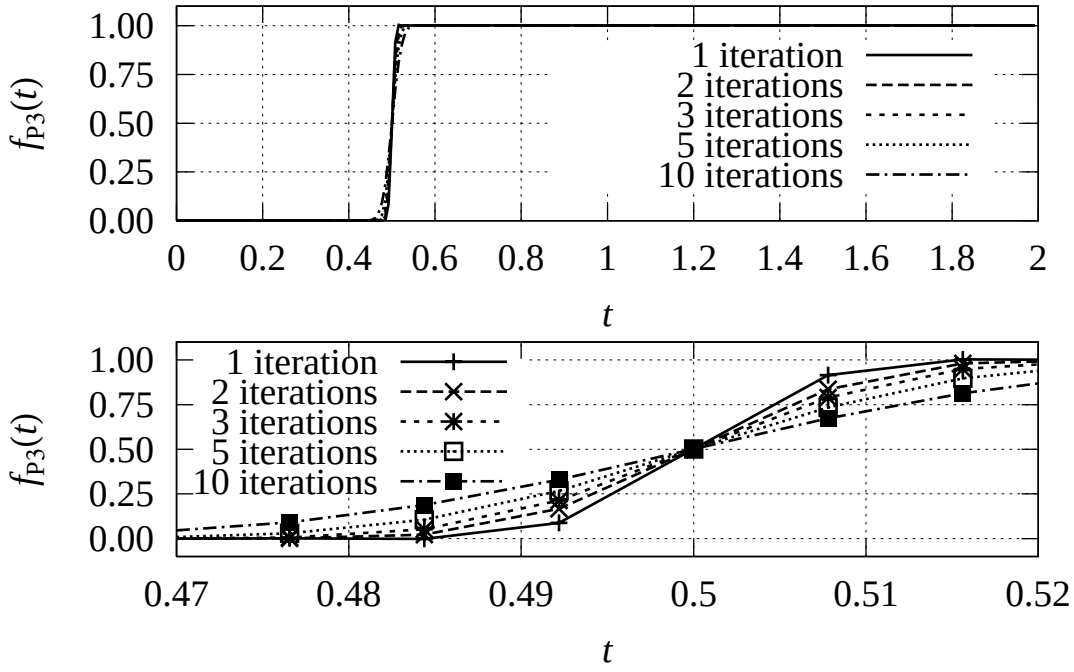


Figure 4.20: Results of repeated applications of the proposed transform pair to the delayed unit step $F(s) = \mathrm{e}^{-s/2}/s$. ($N = 256, T = 1, a = 3.5, i = 3, d = 32$)

These functions satisfy $F_n(s) = O(1/s)$ when $s \to \infty$ for arbitrary n, and thus they are those kinds of functions that yield large truncation errors.

Figures 4.22~4.27 show the filtering results obtained by the conventional transform pair, the no-interpolation pair, and the proposed pair with $N = 256, 2048$. The first two transform pairs give almost the same accuracy. The proposed transform pair reduces the errors, and at the same time, extends the valid region. For example, the valid region is 1.875 times longer in Fig.4.27.

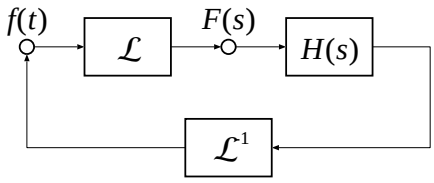


Figure 4.21: The filtering loop.

4.7 Concluding Remarks

In this section, we analyzed interpolations employed in the conventional Laplace transform pair, and showed that the interpolations correspond to extrapolations in the other domain and that the extrapolated data points are discarded in the subsequent stage. Then, we eliminated the interpolations by filling the latter half of $f_{\mathrm{P}}(n\Delta t)$ with zeros and that of $F_{\mathrm{P}}(a + \mathrm{j}k\Delta\omega)$ with the reversed complex conjugate of the former half. The no-interpolation pair turned out to make a numerically reversible pair if $f_{\mathrm{P}}(t)$ is given first and $f_{\mathrm{P}}(0) = 0$. To reduce the errors and to extend the valid region, we applied the proposed inversion to the no-interpolation pair. As a result, the valid region became longer by 1.875 times for $N = 2048$ in the filtering example.

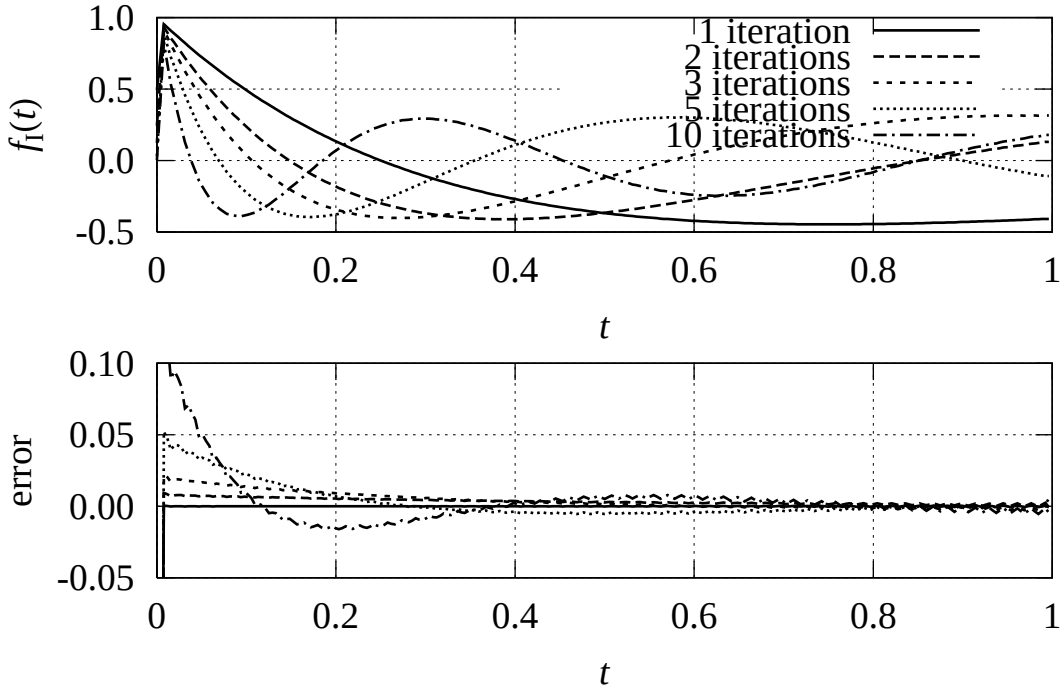


Figure 4.22: The filtering results by the conventional transform pair. ($N = 256, T = 1, a = 3.5$)

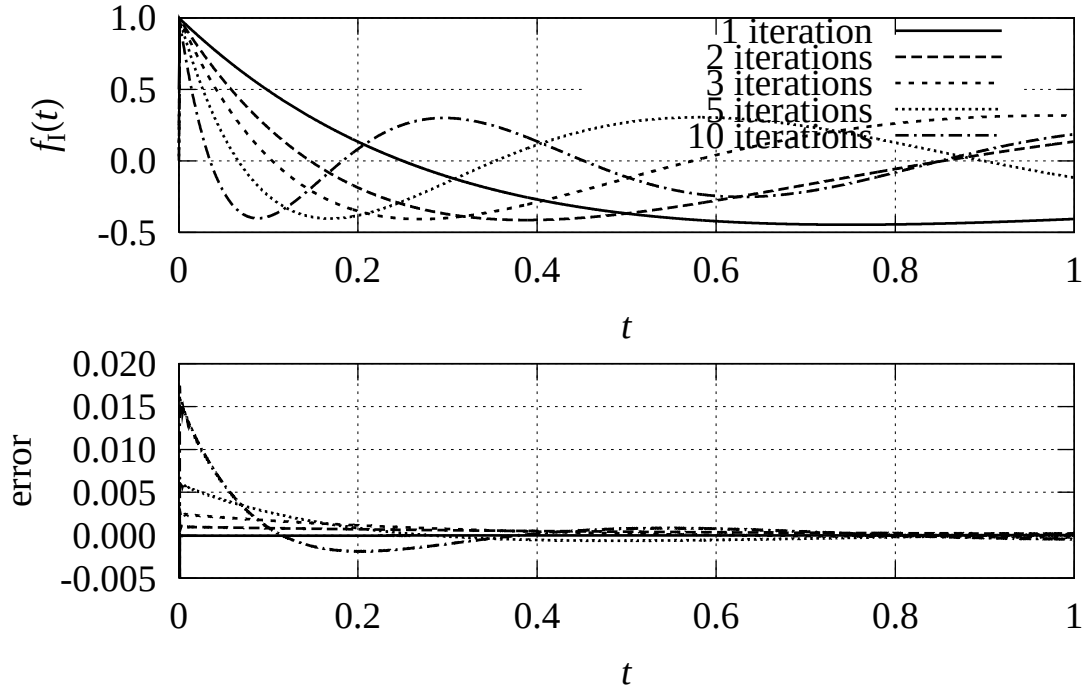


Figure 4.23: The filtering results by the conventional transform pair. ($N = 2048, T = 1, a = 3.5$)

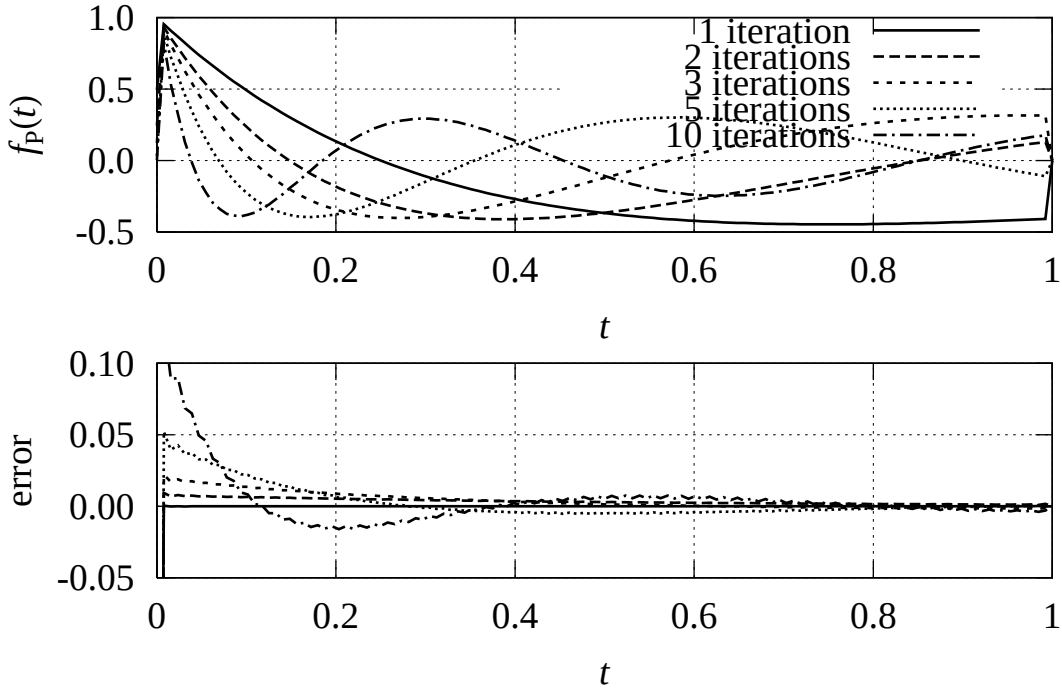


Figure 4.24: The filtering results by the no-interpolation transform pair. ($N = 256, T = 1, a = 3.5$)

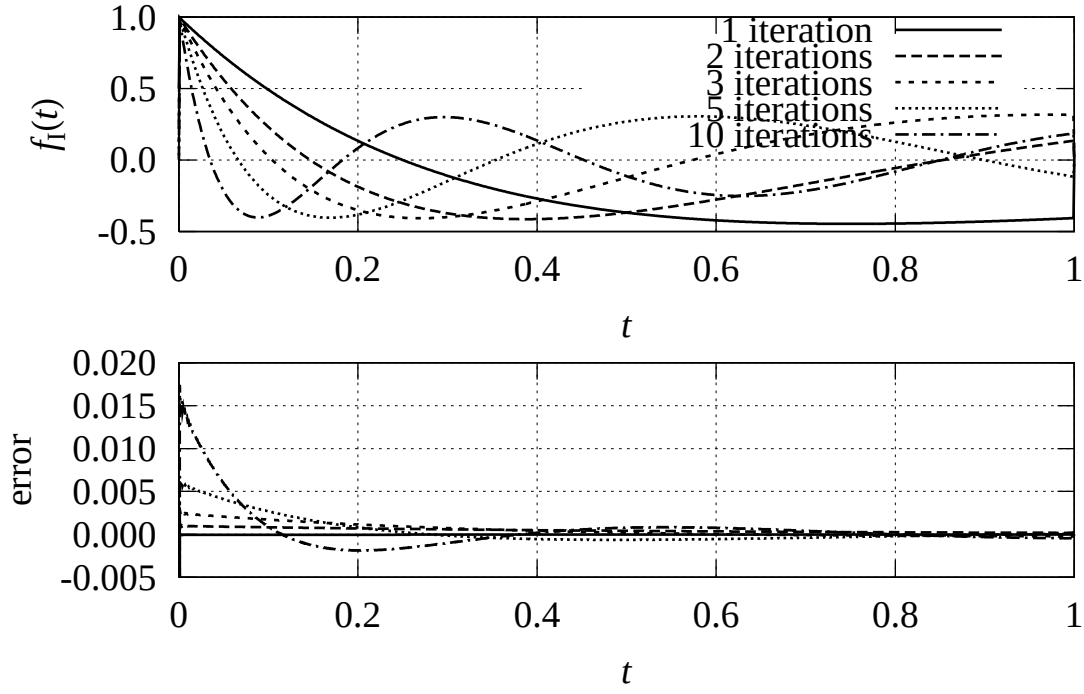


Figure 4.25: The filtering results by the no-interpolate transform pair. ($N = 2048, T = 1, a = 3.5$)

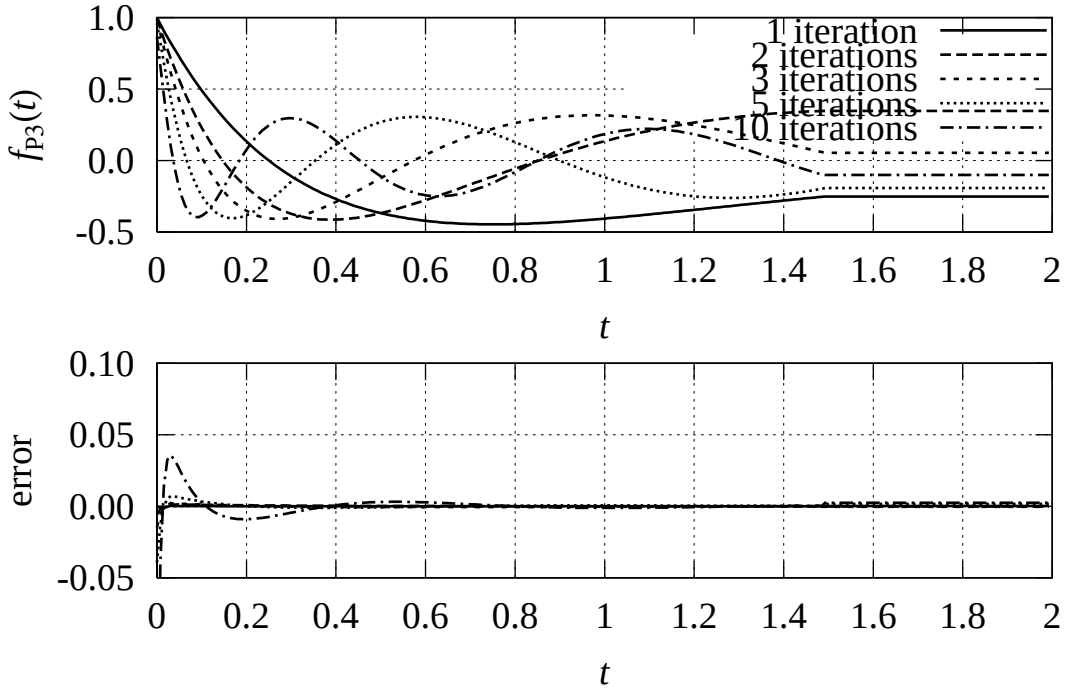


Figure 4.26: The filtering results by the proposed transform pair. ($N = 256, T = 1, a = 3.5, i = 3, d = 64$)

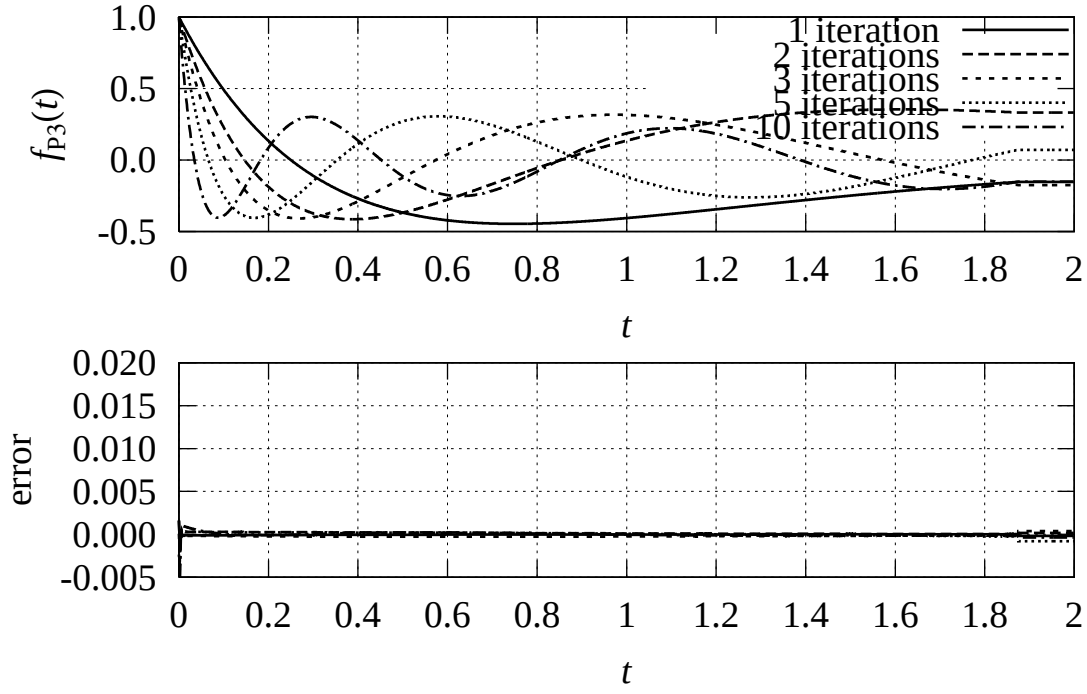


Figure 4.27: The filtering results by the proposed transform pair. ($N = 2048, T = 1, a = 3.5, i = 3, d = 128$)

Chapter 5

FPGA Implementation of the FFT-Based Method

5.1 Introduction

An extensive number of implementations of the FFT algorithm have been reported for various kind of hardware from LSI to recent DSP and FPGA. This is because the FFT algorithm has wide applicability not only to spectrum analysis and frequency-domain filtering, but also to efficient computation of convolution, multiprecision multiplication, and so on. Moreover, the fact that those applications are often in need of realtime data processing promotes their hardware implementations. To deal with transient signals, however, Laplace transforms are more suitable since discrete Fourier transforms (DFT) are theoretically a technique to treat periodic signals. However, no hardware implementation of Laplace transforms has been proposed thus far.

In order to realize realtime systems with Laplace transforms, we implement a Laplace transform processor on an FPGA. We adopt the FFT-based method to enable a transform pair, that is considered to be an indispensable function in the actual use. In the course of the implementation, it turns out that the implementation of the FFT-based inversion with the fixed-point numbers makes the precision of obtained results unacceptably low due to characteristics of Laplace transforms. Although an implementation using floating-point numbers evidently solves the problem, it requires an excessive number of logic gates and becomes slower. Instead of the floating-point numbers, we propose the use of block-floating-point numbers in the FFT block [19] taking into account its butterfly structure.

5.2 Design Specifications

We implement the FFT-based inverse and forward transformations in one design. Referring to Eqs. (2.14) and (2.26), we define the formulas as

$$f_{\mathrm{D}}(n\Delta t) = \frac{\mathrm{e}^{an\Delta t}}{T}\left\{\mathrm{Re}\left[\sum_{k=0}^{N-1} F(a+\mathrm{j}k\Delta\omega)\mathrm{e}^{\mathrm{j}2\pi kn/N}\right] - \frac{1}{2}F(a)\right\}, \tag{5.1}$$

$$F_{\mathrm{D}}(a+\mathrm{j}k\Delta\nu) = \frac{\pi}{\Omega}\left[\sum_{n=0}^{N-1} f(n\Delta\tau)\mathrm{e}^{-an\Delta\tau}\mathrm{e}^{-\mathrm{j}2\pi kn/N} - \frac{1}{2}f(0)\right]. \tag{5.2}$$

These are one-way transformations, that is, they do not make a transform pair. We can yet easily modify them to be the no-interpolation transform pair given by Eqs. (4.29) and (4.31).

We first design the processor using 16-bit fixed-point numbers. The numbers are of 2's complement representation, and the decimal point is placed at right after the first sign bit as follows:

$$b_0.b_1b_2\cdots b_{M-1}, \tag{5.3}$$

where b_0 is the sign bit and $M = 16$ is the word length. This notation can represent numbers from -1 to slightly less than 1.

Figure 5.1 illustrates the block diagram of the Laplace transform processor. The circuit is downloaded into the FPGA device implemented on a PCI board so that the control and data transfer is carried out via the PCI interface of a standard PC/AT compatible computer. Before a transformation, the data $F(a+\mathrm{j}k\Delta\omega)$ or $f(n\Delta\tau)$ to be transformed are stored in the data memory. In the case of an inverse transformation, the FFT block first computes the inverse discrete Fourier transform (IDFT) of the data using the sin/cos table. That is, it computes the summation in Eq. (5.1), and writes back the result to the data memory. Right after the IFFT computation, the Laplace block subtracts $F(a)/2$ from the real parts of the result, multiplies them with $\mathrm{e}^{an\Delta t}$, and writes back the final result to the data memory. The exponential coefficients are computed in advance and stored in the lower 16 bits of the exp table. In the case of a forward transformation, the Laplace block first multiplies the data with $\mathrm{e}^{-an\Delta\tau}$ and write them to the data memory. The exponential coefficients are stored in the higher 16 bits of the exp table. The FFT block then computes DFT of the data, and finally, the Laplace block subtracts $f(0)/2$ from the data.

It was found out that a 16-bit multiplier cannot finish one multiplication in a PCI clock on the device we used. Hence, we designed a pipeline multiplier and shared one multiplier between the FFT block and the Laplace block to reduce logic gates taking into account that they do not operate at the same time.

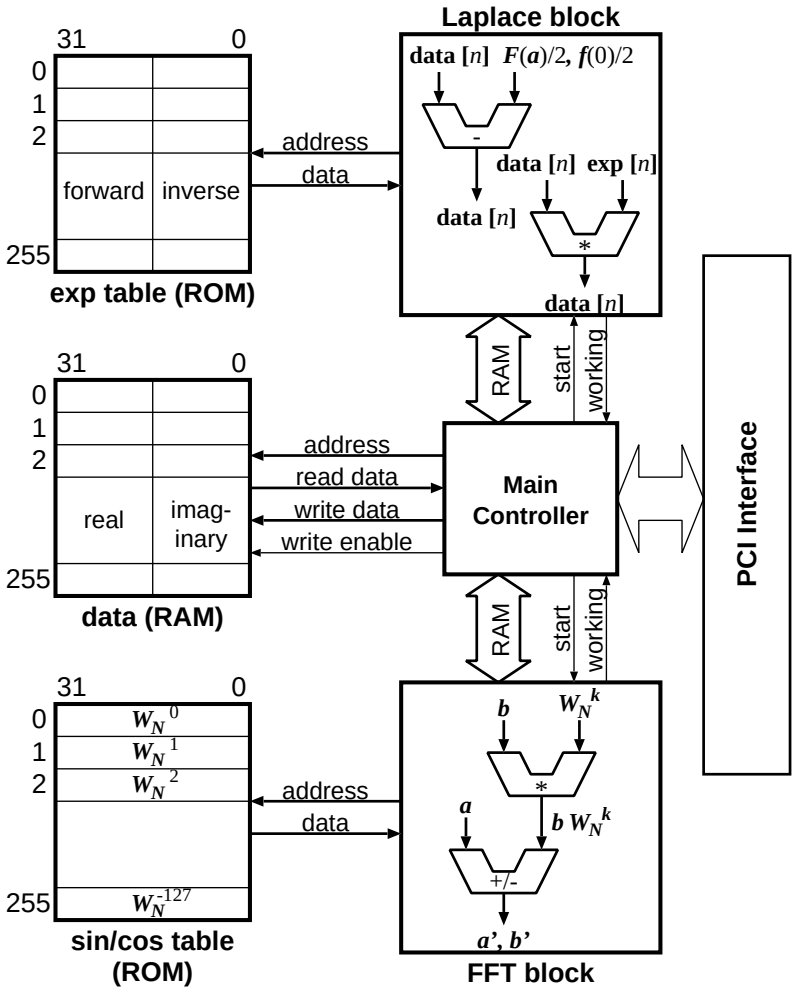


Figure 5.1: The block diagram of the designed Laplace transform processor.

5.3 FFT Block

In the FFT block, we employ the radix-2 decimation-in-time algorithm whose butterfly structure is shown in Fig. 5.2. The FFT block has an arithmetic unit that calculates the constituent element of the butterfly structure illustrated in Fig. 5.3:

$$a' = \frac{1}{2}\left(a + bW_N^k\right), \quad b' = \frac{1}{2}\left(a - bW_N^k\right), \tag{5.4}$$

where $W_N = \mathrm{e}^{-\mathrm{j}2\pi/N}$ is the twiddle factor. The addition/subtraction of a and bW_N^k may cause an overflow since variables are represented by the fixed-point numbers, Eq. (5.3). By dividing

$a \pm bW_N^k$ by two, we have $|a'| < 1, |b'| < 1$ if $|a| < 1, |b| < 1$, and prevent an overflow. The division by two is implemented by a 1-bit right shift followed by rounding to the nearest number. For an N-point FFT, the arithmetic unit repeats the butterfly calculation $(N/2)\log_2 N$ times in total, stage by stage.

Thus, for an input sequence $x(n), n = 0, 1, \cdots, N-1$, the FFT block outputs its DFT given by

$$X(k) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)W_N^{nk}, \quad N = 2^{d_N}, \tag{5.5}$$

where $k = 0, \cdots, N-1$ and d_N denotes $\log_2 N$ that is equal to the number of the butterfly stages. In the computation, only the former half of the sin/cos table is addressed since Eq. (5.4) makes use of the identity $W_N^{k+N/2} = -W_N^k$. By preparing $W_N^{-k}, k = 0, 1, \cdots, N/2-1$ in the latter half of the sin/cos table, the FFT block also realizes IDFT of $X(k), k = 0, 1, \cdots, N-1$, given by

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)W_N^{-kn}. \tag{5.6}$$

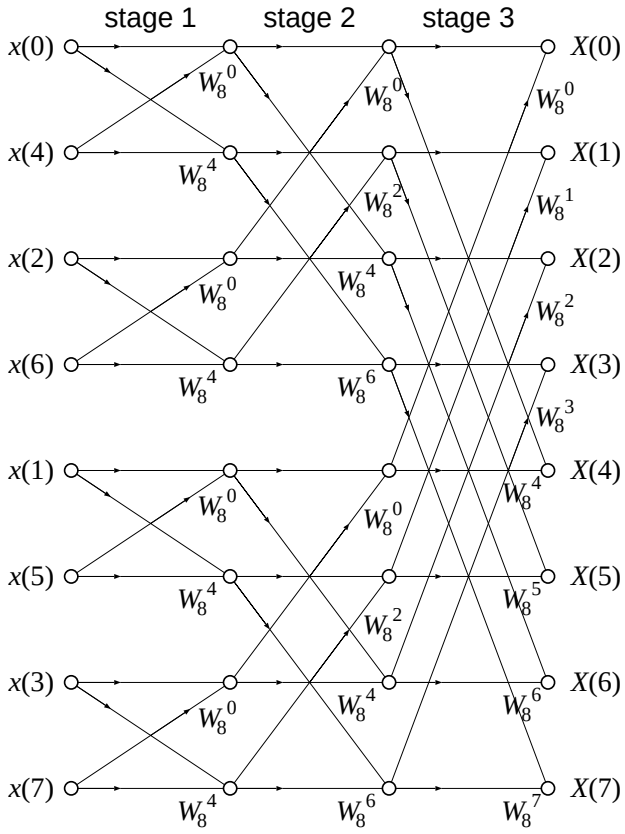


Figure 5.2: The decimation-in-time FFT algorithm. ($N = 8$)

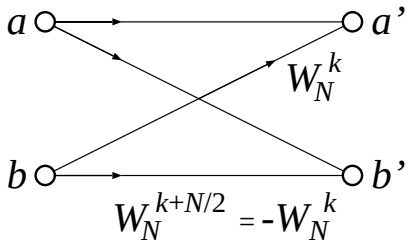


Figure 5.3: The butterfly element.

In addition, in the sin/cos table, some of the values are exactly equal to one which Eq. (5.3) is not able to express. Hence, in the table, we placed the decimal point at after the second bit as

$$b_0 b_1 . b_2 \cdots b_{M-1}, \quad b_0 : \text{the sign bit.} \tag{5.7}$$

5.4 Laplace block

Inverse transformation After the FFT block computes the IDFT of given $F(a + \mathrm{j}k\Delta\omega)$, the Laplace block completes the remaining computations in Eq. (5.1). That is:

1. Subtract $F(a)/2N$ from the real part of each entry of the data memory,
2. Multiply the result by $\mathrm{e}^{a(n-N/2)\Delta t}$ and write it back to the data memory.

These computations are executed for only former half $n = 0, 1, \cdots, N/2 - 1$. Namely, we discard the latter half since it contains the large truncation errors as stated in Section 3.2.

In the step 1, we subtract $F(a)/2N$, instead of $F(a)/2$, since the FFT block outputs IDFT of $F(a + \mathrm{j}k\Delta\omega)$ divided by N as Eq. (5.6). In the step 2, the exponential coefficients are scaled down so that the maximum value of $\mathrm{e}^{-a\Delta t}$ at $n = N/2 - 1$ falls in the range representable with Eq. (5.3). Thus, one obtains the scaled output $K_{\mathrm{inv}} f_{\mathrm{D}}(n\Delta t)$ for $n = 0, \cdots, N/2-1$ with the scaling coefficient K_{inv} defined by

$$K_{\mathrm{inv}} = \frac{1}{N} \cdot \mathrm{e}^{-a(N/2)\Delta t} \cdot T = \frac{T}{N\mathrm{e}^{aT}}. \tag{5.8}$$

Forward transformation A forward Laplace transformation of $f(t)$ proceeds as follows:

1. Multiply each entry of the data memory by $\mathrm{e}^{-an\Delta\tau}$ and write it back,
2. Compute the DFT of the data memory,
3. Subtract $f(0)/2N$ from each entry of the data and write it back.

In the step 1, the multiplication is just skipped for $n = 0$ since $\mathrm{e}^{-a\cdot 0\cdot\Delta\omega} = 1$ and that cannot even be represented by Eq. (5.3). The step 3 is executed for only $k = 0, 1, \cdots, N/2$, since the latter half is the reversed complex conjugate of the former half. In the step 3, we subtract $f(0)/2N$, instead of $f(0)/2$, by the same reason as the inversion. Thus, one obtains the scaled output $K_{\mathrm{fwd}} F_{\mathrm{D}}(a+\mathrm{j}k\Delta\nu)$ for $k = 0, 1, \cdots, N/2$ with the scaling coefficient K_{fwd} defined by

$$K_{\mathrm{fwd}} = \frac{1}{N} \cdot \frac{\Omega}{\pi} = \frac{1}{T}. \tag{5.9}$$

5.5 Examples

Figure 5.4 illustrates the process of the Laplace inversion of $F(s) = 3/s$. To see the precision of the intermediate and the final results, the numbers expressed in the right vertical axis show the values when viewed as 16-bit integers, namely, they show

$$-2^{M-1}b_0 + \sum_{i=1}^{M-1} 2^{M-1-i}b_i = -2^{M-1}b_0 + 2^{M-2}b_1 + \cdots 2b_{M-2} + b_{M-1}. \tag{5.10}$$

With $N = 256$, $T = 1$, and $a = 3.5$, we have $K_{\text{inv}} = 0.000118$ and the expected result is then

$$K_{\text{inv}} f(t) = 3K_{\text{inv}} = 0.000354 = 11.6/32768. \tag{5.11}$$

The right bottom figure shows that we seemingly have the expected result, however, with merely 4-bit precision and accordingly with low accuracy, though given $F(s)$ has 16-bit precision. Figures 5.5 and 5.6 show examples for $F(s) = 70/(s+5)^2$ and $F(s) = 8\omega/((s+1)^2 + \omega^2), \omega = 8\pi$, respectively. The results have merely 5- or 6-bit precision starting with $F(s)$ of 16-bit precision.

There are two reasons for the worse precision. The first reason is that a Laplace transform usually has poor content in its magnitude as shown in the upper left figure of Fig. 5.4 and the left figure of Figs. 5.5 and 5.6: the magnitude content concentrates at low frequencies and rapidly decreases at high frequencies. Since the IDFT conserves the squared norm as stated by the Parseval's theorem, the fact implies that the obtained values through an IFFT is small especially for uniformly distributed functions such as the step function. The second reason is the wide dynamic range of the exponential function. The exponential coefficients cancel out the exponential decrease in $\text{Re}[\text{IFFT}[F(s)]] - F(a)/2$. The dynamic range is e^{aT} and thereby, for the step function for instance, the final result cannot be larger than

$$\frac{2^{M-1}}{\mathrm{e}^{aT}} = 990, \tag{5.12}$$

when viewed as 16-bit integers as Eq. (5.10). Namely, the wide dynamic range limits the precision of the inversion of the function to $(M-1) - aT/\log 2 \simeq 10$ bits. This is an inevitable difficulty when implementing the Laplace inversion using the fixed-point numbers.

In the next Fig. 5.7 is illustrated the process of the Laplace transformation of $f(t) = 0.99$, and Figs. 5.8 and 5.9 show examples for $f(t) = 13t\mathrm{e}^{-5t}$ and $f(t) = \mathrm{e}^{-t}\sin\omega t, \omega = 8\pi$, respectively. The results retain 13- or 14-bit precision starting with $f(t)$ of 16-bit precision. The situation is fairly better than that of the inversion since $F_{\text{D}}(s)$ is usually enough peaky to keep the precision through the FFT, and in addition, the exponential function merely attenuates $f(t)$.

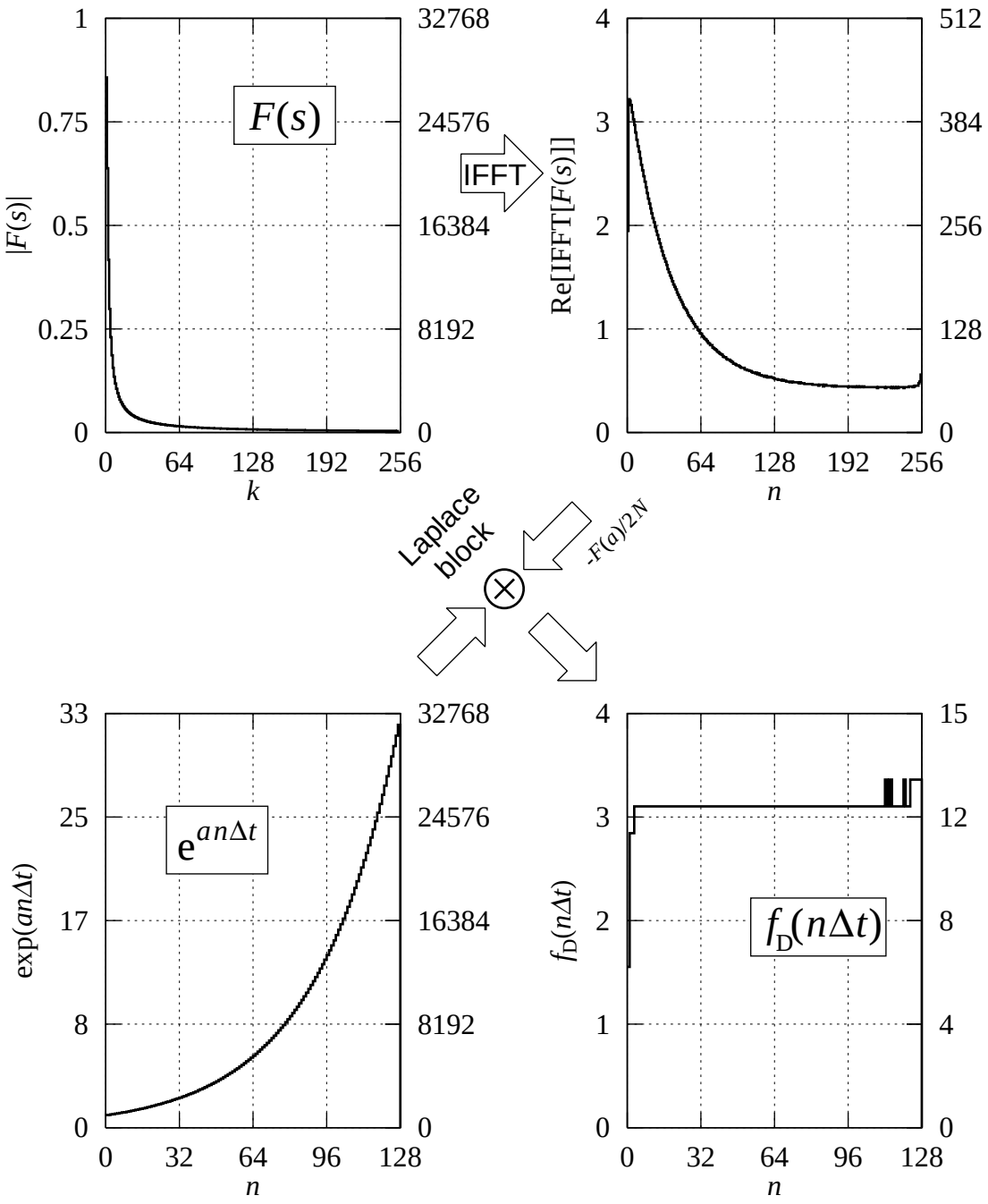


Figure 5.4: The process of Laplace inversion with fixed-point numbers for the step function $F(s) = 3/s$. ($N = 256, T = 1, a = 3.5, K_{\text{inv}} = 0.000118$)

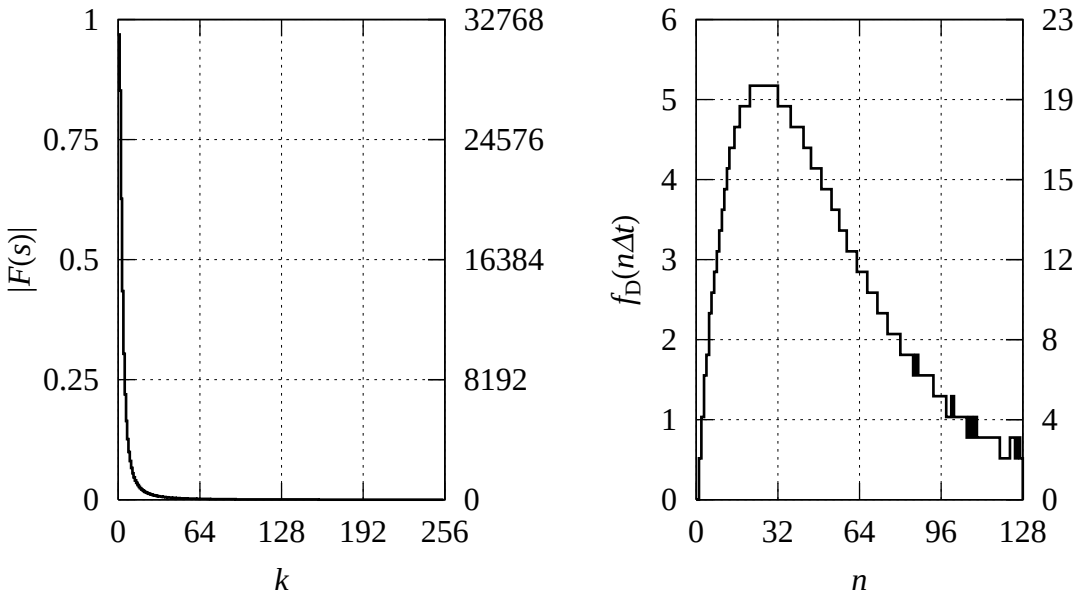


Figure 5.5: The Laplace inversion of $F(s) = 70/(s+5)^2$ with fixed-point numbers. ($\mathcal{L}^{-1}[F](t) = 70te^{-5t}, N = 256, T = 1, a = 3.5, K_{\text{inv}} = 0.000118$)

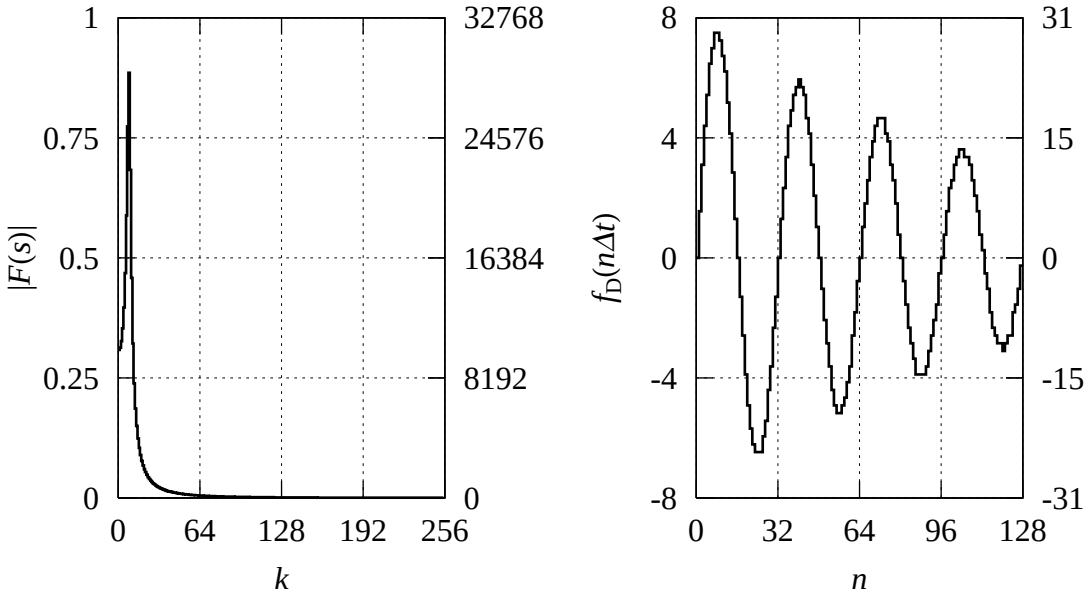


Figure 5.6: The Laplace inversion of $F(s) = 8\omega/((s+1)^2+\omega^2), \omega = 8\pi$ with fixed-point numbers. ($\mathcal{L}^{-1}[F](t) = 8e^{-t}\sin\omega t, N = 256, T = 1, a = 3.5, K_{\text{inv}} = 0.000118$)

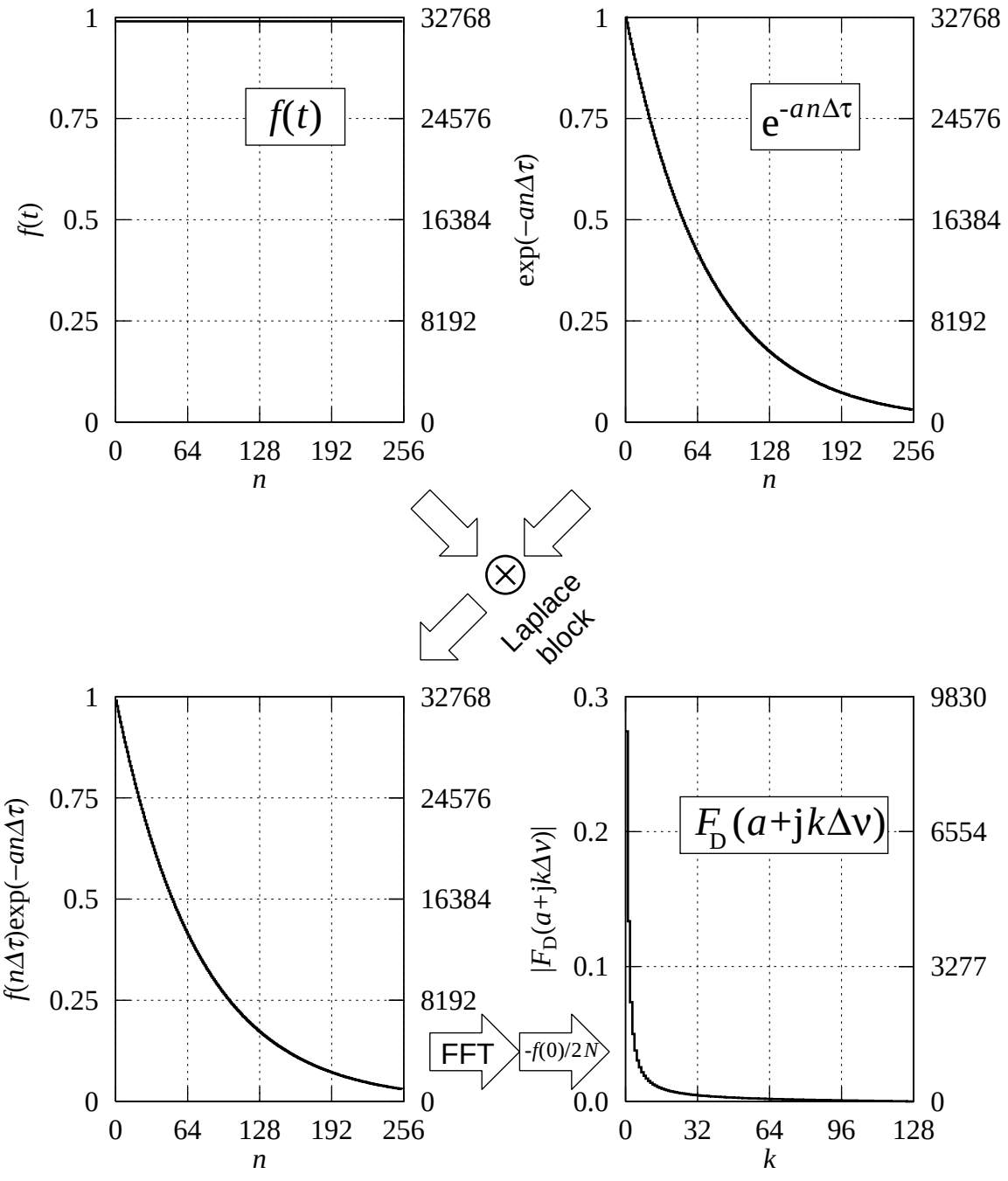


Figure 5.7: The process of Laplace transformation with fixed-point numbers for the step function $f(t) = 0.99$. ($N = 256, T = 1, a = 3.5, K_{\text{fwd}} = 1$)

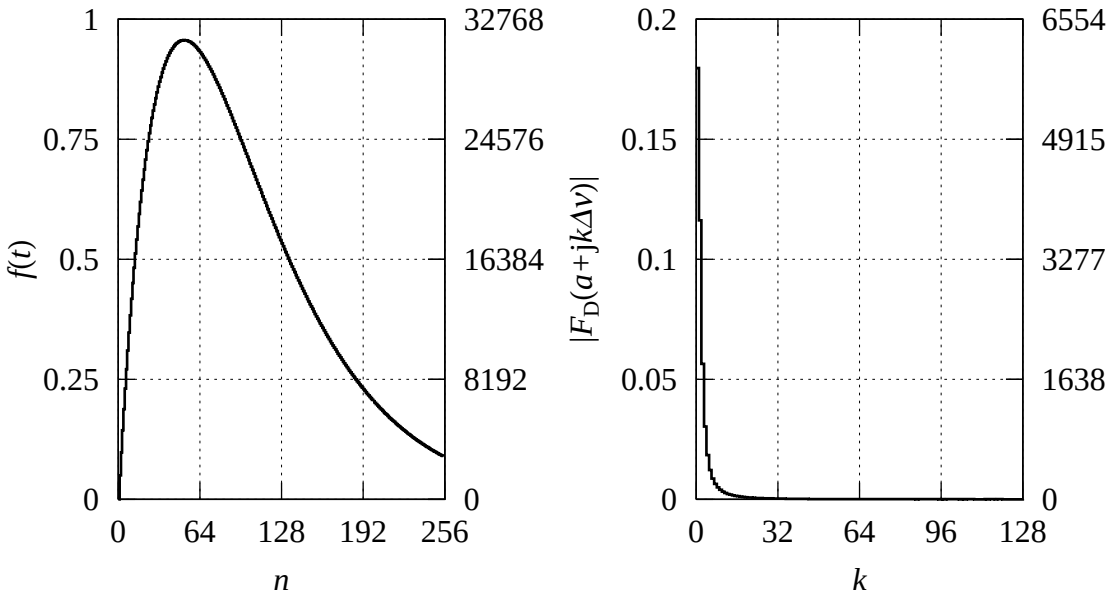


Figure 5.8: The Laplace transform of $f(t) = 13te^{-5t}$ with fixed-point numbers. ($\mathcal{L}[f](s) = 13/(s+5)^2, N = 256, T = 1, a = 3.5, K_{\text{fwd}} = 1$)

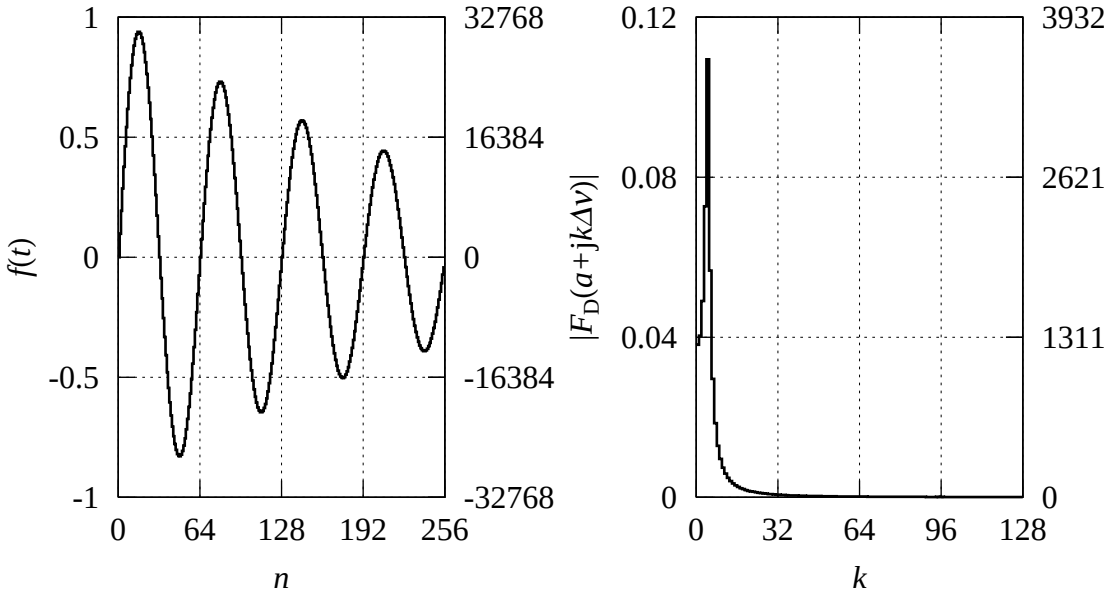


Figure 5.9: The Laplace transform of $f(t) = e^{-t}\sin\omega t, \omega = 8\pi$ with fixed-point numbers. ($\mathcal{L}[f](s) = \omega/((s+1)^2 + \omega^2), N = 256, T = 1, a = 3.5, K_{\text{fwd}} = 1$)

5.6 Introducing Block-Floating-Point Numbers

To improve the precision especially for the inversion, we consider to remedy the degradation of precision incurred by the FFT block. Since the magnitude content of $F(s)$ is usually poor especially at high frequencies, an overflow is unlikely to occur frequently in the calculation of $a \pm bW_N^k$. In addition, the butterfly structure shown in Fig. 5.2 allows us to choose to divide the result of $a \pm bW_N^k$ by two or not stage by stage provided that the choice is consistent within each stage. Therefore, we can eliminate unnecessary divisions by two in a stage if there is no possibility of an overflow in the stage.

We derive here a criterion, used during computation of each stage, of the possibility of an overflow in the subsequent stage. From the triangle inequality of complex numbers, we have

$$|\mathrm{Re}[a \pm bW_N^k]| \le |a \pm bW_N^k| \le |a| + |b|, \tag{5.13}$$

and the same inequality holds for $|\mathrm{Im}[a \pm bW_N^k]|$. Hence, for no overflow occurrence, namely

$$|\mathrm{Re}[a \pm bW_N^k]| < 1, \quad |\mathrm{Im}[a \pm bW_N^k]| < 1, \tag{5.14}$$

the following condition is sufficient:

$$|a| < \frac{1}{2}, \quad |b| < \frac{1}{2}. \tag{5.15}$$

However, calculation of the absolute value of a complex number requires costly computation of square of the real and imaginary part. To avoid the square computation, we use a simplified sufficient criterion

$$|\mathrm{Re}[a]| < \frac{1}{2\sqrt{2}}, \quad |\mathrm{Im}[a]| < \frac{1}{2\sqrt{2}}, \quad |\mathrm{Re}[b]| < \frac{1}{2\sqrt{2}}, \quad |\mathrm{Im}[b]| < \frac{1}{2\sqrt{2}} \tag{5.16}$$

which is more suitable for hardware implementation.

Using the above criterion, we modify the FFT computation as follows.

1. $m := 1, \quad c := 2, \quad d := 1$

2. Execute all the butterfly calculations in the m-th stage using

$$a' = \frac{1}{c}\left(a + bW_N{}^k\right), \quad b' = \frac{1}{c}\left(a - bW_N{}^k\right).$$

3. If, in the step 2, all a', b' satisfy

$$|\mathrm{Re}[a']| < \frac{1}{2\sqrt{2}}, \quad |\mathrm{Im}[a']| < \frac{1}{2\sqrt{2}}, \quad |\mathrm{Re}[b']| < \frac{1}{2\sqrt{2}}, \quad |\mathrm{Im}[b']| < \frac{1}{2\sqrt{2}} \tag{5.17}$$

then $c := 1$ or else $c := 2, d := d + 1$.

4. $m := m + 1$. If $m \leq \log_2 N$ then go to the step 2 or else end.

The variable d denotes the number of stages in which the divisions were used.

Thus, the outputs and the scaling factors $K_{\mathrm{inv}}, K_{\mathrm{fwd}}$ are modified as

$$x(n) = \frac{1}{2^d} \sum_{k=0}^{N-1} X(k) W_N^{-nk}, \quad K'_{\mathrm{inv}} = \frac{1}{2^d \mathrm{e}^{aT}}, \tag{5.18}$$

$$X(k) = \frac{1}{2^d} \sum_{n=0}^{N-1} x(n) W_N^{nk}, \quad K'_{\mathrm{fwd}} = \frac{N}{2^d T} = \frac{2^{d_N - d}}{T}. \tag{5.19}$$

The improvement achieved by the modification is up-scaling by the factor $N/2^d = 2^{d_N - d}$, which is dependent on the original function. This modification is equivalent to introducing block-floating-point numbers [19] to the FFT block, and regarding each stage as a block. From this viewpoint, the place of the decimal point is changed stage by stage depending on the possibility of an overflow. That is, the division by two corresponds to a change of the decimal point one to the right.

Examples Figure 5.10 shows the proposed result of the Laplace inversion of $F(s) = 3/s$. For this case, d was 3 and thereby the result is improved by the factor $2^{d_N - d} = 32$. The improved scaling factor K'_{inv} and the expected result are

$$K'_{\mathrm{inv}} = 0.00377, \tag{5.20}$$

$$K'_{\mathrm{inv}} f(t) = 3K'_{\mathrm{inv}} = 0.0113 = 371/32768. \tag{5.21}$$

The right figure shows that the expected result is obtained, and the precision is improved from 4 bits to 9 bits. The precision approached the limit Eq. (5.12) within about 1.4 bits. Figs. 5.11 and 5.12 show improved results for the examples presented in Figs. 5.5 and 5.6, where the precision is improved by 4 bits and 5 bits, respectively. Consequently, the rugged errors due to the lack of precision disappeared.

As for the Laplace transformation, Figs. 5.13~5.15 show the proposed results for the examples presented in Figs. 5.7~5.9. The precision is improved by 1 bit for the first two examples and by 2 bits for the last one, to within about 1 bit to the full 16-bit precision.

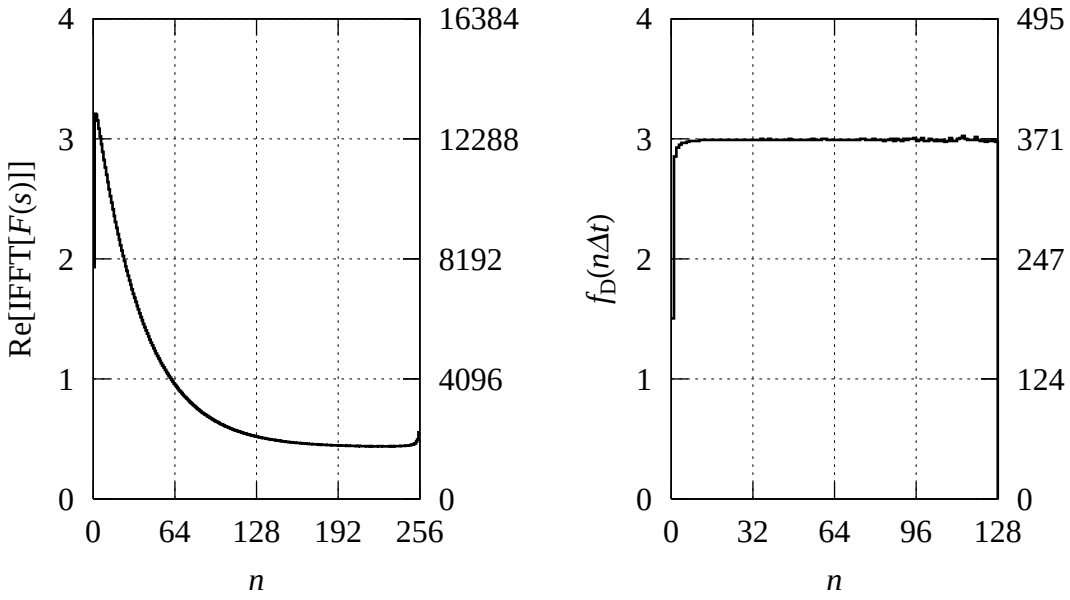


Figure 5.10: The process of Laplace inversion with block-floating-point numbers for the step function $F(s) = 3/s$. ($N = 256, T = 1, a = 3.5, d = 3, K'_{\text{inv}} = 0.00377$)

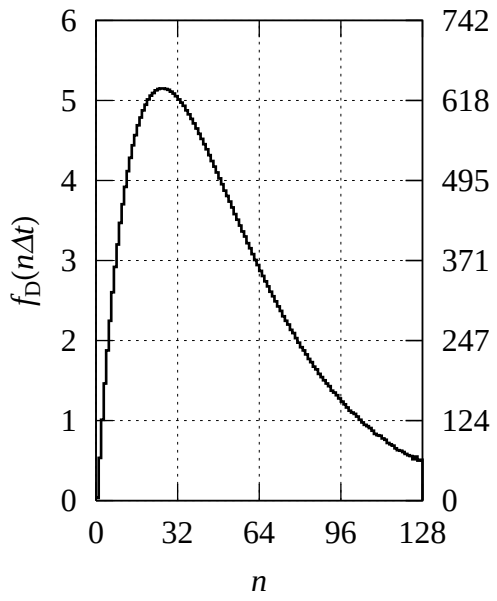


Figure 5.11: The Laplace inversion of $F(s) = 70/(s + 5)^2$ with block-floating-point numbers. ($\mathcal{L}^{-1}[F](t) = 70te^{-5t}, N = 256, T = 1, a = 3.5, d = 4, K'_{\text{inv}} = 0.00189$)

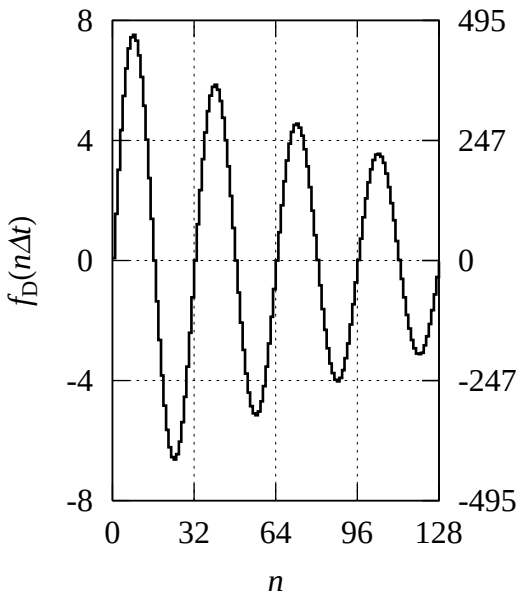


Figure 5.12: The Laplace inversion of $F(s) = 8\omega/((s + 1)^2 + \omega^2), \omega = 8\pi$ with block-floating-point numbers. ($\mathcal{L}^{-1}[F](t) = 8e^{-t} \sin \omega t, N = 256, T = 1, a = 3.5, d = 3, K'_{\text{inv}} = 0.00377$)

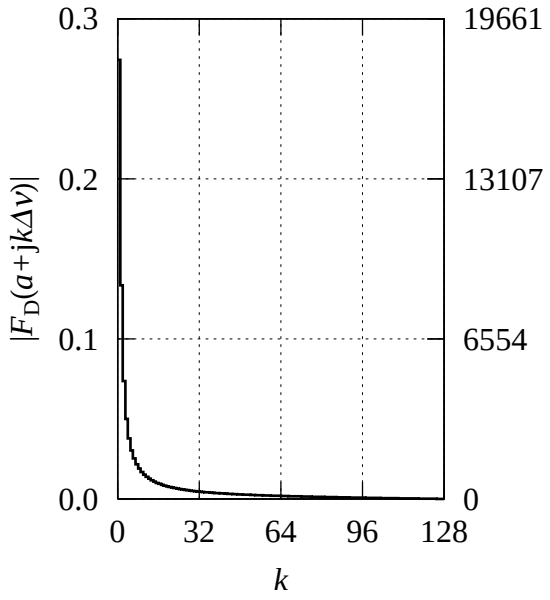


Figure 5.13: The Laplace transform of the step function $f(t) = 0.99$ with block-floating-point numbers. ($N = 256, T = 1, a = 3.5, d = 7, K'_{\text{fwd}} = 2$)

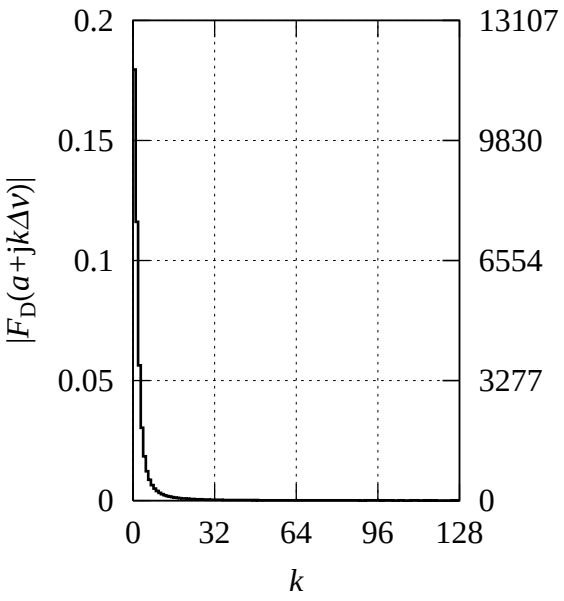


Figure 5.14: The Laplace transform of $f(t) = 13te^{-5t}$ with block-floating-point numbers. ($\mathcal{L}[f](s) = 13/(s+5)^2, N = 256, T = 1, a = 3.5, d = 7, K'_{\text{fwd}} = 2$)

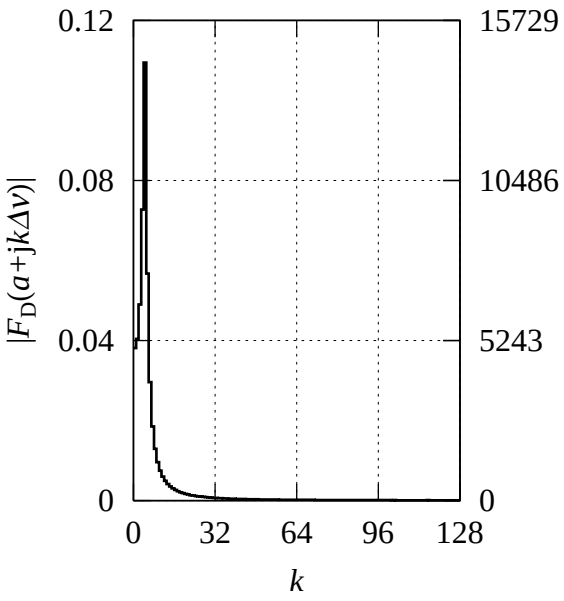


Figure 5.15: The Laplace transform of $f(t) = e^{-t}\sin\omega t, \omega = 8\pi$ with block-floating-point numbers. ($\mathcal{L}[f](s) = \omega/((s+1)^2 + \omega^2), N = 256, T = 1, a = 3.5, d = 6, K'_{\text{fwd}} = 4$)

5.7 Design Result

We implemented the designed processor on a 100,000-gate FPGA device, EPF10K100ARC240-1 manufactured by Altera using VHDL for its hardware description. The result of placing and routing is shown in Table 5.1 for the implementation using fixed-point numbers, and in Table 5.2 for the implementation using block-floating-point numbers.

At the PCI clock frequency, 33 MHz, it takes 134 μs to invert a Laplace transform and 149 μs to transform a time function.

Table 5.1: The design result for implementation using fixed-point numbers

Maximum Clock Frequency	36.49MHz
Logic Element usage	38% (38,140 gates)
Embedded Array Block usage	100% (24Kb)

Table 5.2: The design result for implementation using block-floating-point numbers

Maximum Clock Frequency	35.58MHz
Logic Element usage	39% (39,060 gates)
Embedded Array Block usage	100% (24Kb)

5.8 Concluding Remarks

We presented an implementation of the FFT-based numerical Laplace transforms on an FPGA. It turned out that the implementation with fixed-point numbers results in the severe lack of precision, and consequently in large rugged errors especially for the Laplace inversion. One of the reasons is the poor magnitude content of $F(s)$ that makes the IFFT result unacceptably small. The other reason is the wide dynamic range of the exponential function, which is inherent in Laplace transforms and limits the precision of inversion. By introducing the block-floating-point numbers to the FFT computation, we eliminated unnecessary divisions by two and improved the precision that had been reduced through the IFFT. Since we can still use fixed-point arithmetics, the modification was achieved with a slight increase of the complexity of the circuit. The precision was improved by several bits to within $1 \sim 2$ bits to the inherent limit in the case of inverse transformations. It was also improved for forward transformations by a few bits to within about 1 bit to the full 16-bit precision.

Chapter 6

Application to Analysis of Transmission Lines

6.1 Introduction

In the following two chapters, we present transient analysis of power transmission lines at fault occurrence. Generally in transient analysis of transmission lines including high speed interconnects in LSIs, evaluation of waveform degradation caused by frequency dependent losses, such as skin effect, is of great importance. To deal with the frequency dependent losses in the transient analysis, Laplace transforms are the most suitable since treatment of frequency dependent terms in the time domain requires computation of convolutions that are computationally expensive. In addition, complexity of their expressions in the form of Laplace transforms makes it difficult to invert them analytically. Therefore, numerical Laplace transforms are necessary to evaluate the frequency dependent losses.

This chapter presents fundamentals of analysis of transmission lines in the complex frequency domain covering matrix form of the telegraph equation for multi-phase transmission lines, ground loss effects, and calculation of matrix functions.

6.2 Telegraph Equation

Figure 6.1 illustrates an n-phase transmission line above a ground in which $v_k(x,t), i_k(x,t)$ are respectively the k-th phase voltage and the k-th phase current at position x from the left end, and at time t. Supposing that the line does not have frequency dependent losses, the telegraph

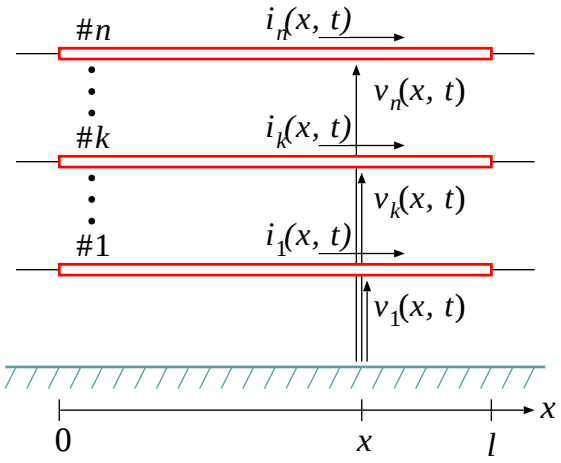


Figure 6.1: An n-phase transmission line.

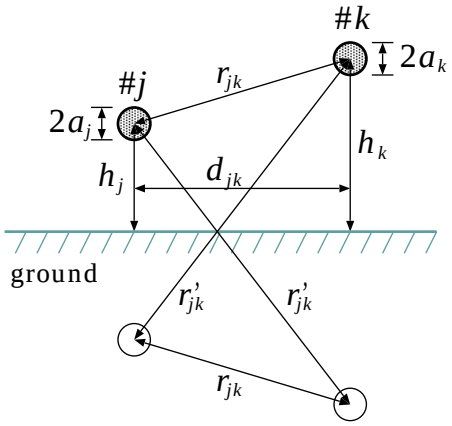


Figure 6.2: Configuration of conductors #j, #k.

equation is given by

$$-\frac{\partial \boldsymbol{v}}{\partial x} = \boldsymbol{L}\frac{\partial \boldsymbol{i}}{\partial t} + \boldsymbol{R}\boldsymbol{i}, \quad -\frac{\partial \boldsymbol{i}}{\partial x} = \boldsymbol{C}\frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{G}\boldsymbol{v}, \tag{6.1}$$

where $\boldsymbol{v}$, $\boldsymbol{i}$ are column vectors of order n defined by (T denotes transpose)

$$\boldsymbol{v} = \begin{pmatrix} v_1 & v_2 & \cdots & v_n \end{pmatrix}^T, \quad \boldsymbol{i} = \begin{pmatrix} i_1 & i_2 & \cdots & i_n \end{pmatrix}^T. \tag{6.2}$$

The $n \times n$ matrices $\boldsymbol{L}, \boldsymbol{C}$ represent inductance and capacitance of the line per unit length, respectively. With the geometry of conductors shown in Fig. 6.2, they are given by [20, 21]

$$L_{jj} = \frac{\mu_0}{2\pi}\log\frac{2h_j}{a_j}, \qquad L_{jk} = L_{kj} = \frac{\mu_0}{2\pi}\log\frac{r'_{jk}}{r_{jk}}, \tag{6.3}$$

$$S_{jj} = \frac{1}{2\pi\varepsilon_0}\log\frac{2h_j}{a_j}, \qquad S_{jk} = S_{kj} = \frac{1}{2\pi\varepsilon_0}\log\frac{r'_{jk}}{r_{jk}}, \tag{6.4}$$

$$r_{jk} = \sqrt{(h_j - h_k)^2 + {d_{jk}}^2}, \qquad r'_{jk} = \sqrt{(h_j + h_k)^2 + {d_{jk}}^2}, \tag{6.5}$$

where $\boldsymbol{S} = \boldsymbol{C}^{-1}$. The diagonal elements L_{jj}, C_{jj} represent the self inductance and the self capacitance of the conductor #j, and off-diagonal elements $L_{jk}, C_{jk} (j \neq k)$ are the mutual inductance and the mutual capacitance between the conductors #j and #k. The other $n \times n$ matrices $\boldsymbol{R}, \boldsymbol{G}$ respectively represent resistance and conductance of the line per unit length. In addition, these four matrices are symmetric from the reciprocity.

If there are no initial distributions, the Laplace transformation of equation (6.1) yields

$$-\frac{\partial \boldsymbol{V}}{\partial x} = \boldsymbol{Z}(s)\boldsymbol{I}, \quad -\frac{\partial \boldsymbol{I}}{\partial x} = \boldsymbol{Y}(s)\boldsymbol{V}, \tag{6.6}$$

$$\boldsymbol{Z}(s) = s\boldsymbol{L} + \boldsymbol{R}, \quad \boldsymbol{Y}(s) = s\boldsymbol{C} + \boldsymbol{G}, \tag{6.7}$$

where $\boldsymbol{V}$, $\boldsymbol{I}$ denote Laplace transforms of $\boldsymbol{v}$, $\boldsymbol{i}$, respectively. To solve Eq. (6.6), let us consider $\boldsymbol{V} + \boldsymbol{Z}_0(s)\boldsymbol{I}$ where $\boldsymbol{Z}_0(s)$ is an arbitrary square matrix for the time being. Then, we have

$$\begin{aligned}\frac{\partial}{\partial x}[\boldsymbol{V} + \boldsymbol{Z}_0(s)\boldsymbol{I}] &= -\boldsymbol{Z}(s)\boldsymbol{I} - \boldsymbol{Z}_0(s)\boldsymbol{Y}(s)\boldsymbol{V} \\ &= -\boldsymbol{Z}_0(s)\boldsymbol{Y}(s)[\boldsymbol{V} + \boldsymbol{Y}^{-1}(s)\boldsymbol{Z}_0{}^{-1}(s)\boldsymbol{Z}(s)\boldsymbol{Z}_0{}^{-1}(s)\boldsymbol{Z}_0(s)\boldsymbol{I}].\end{aligned} \tag{6.8}$$

Now we determine $\boldsymbol{Z}_0(s)$ such that it satisfies

$$\begin{aligned}&\boldsymbol{Y}^{-1}(s)\boldsymbol{Z}_0{}^{-1}(s)\boldsymbol{Z}(s)\boldsymbol{Z}_0{}^{-1}(s) = \boldsymbol{1}, \\ &\therefore\ \boldsymbol{Z}(s) = \boldsymbol{Z}_0(s)\boldsymbol{Y}(s)\boldsymbol{Z}_0(s),\end{aligned} \tag{6.9}$$

and then Eq. (6.8) becomes

$$\frac{\partial}{\partial x}[\boldsymbol{V} + \boldsymbol{Z}_0(s)\boldsymbol{I}] = -\boldsymbol{Z}_0(s)\boldsymbol{Y}(s)[\boldsymbol{V} + \boldsymbol{Z}_0(s)\boldsymbol{I}]. \tag{6.10}$$

Similarly, the following equation holds with the same $\boldsymbol{Z}_0(s)$:

$$\frac{\partial}{\partial x}[\boldsymbol{V} - \boldsymbol{Z}_0(s)\boldsymbol{I}] = +\boldsymbol{Z}_0(s)\boldsymbol{Y}(s)[\boldsymbol{V} - \boldsymbol{Z}_0(s)\boldsymbol{I}]. \tag{6.11}$$

The solution of the above two equations is given by

$$\boldsymbol{Q}(s) = \boldsymbol{Z}_0(s)\boldsymbol{Y}(s), \tag{6.12}$$

$$\boldsymbol{V} + \boldsymbol{Z}_0(s)\boldsymbol{I} = \exp(-\boldsymbol{Q}(s)x)\boldsymbol{A}(s), \tag{6.13}$$

$$\boldsymbol{V} - \boldsymbol{Z}_0(s)\boldsymbol{I} = \exp(\boldsymbol{Q}(s)x)\boldsymbol{B}(s), \tag{6.14}$$

where $\boldsymbol{A}(s)$ and $\boldsymbol{B}(s)$, column vectors of order n, are the constants of integration with respect to x. Consequently, $\boldsymbol{V}$ and $\boldsymbol{I}$ are given by

$$\boldsymbol{V} = \exp(-\boldsymbol{Q}(s)x)\boldsymbol{A}(s) + \exp(\boldsymbol{Q}(s)x)\boldsymbol{B}(s), \tag{6.15}$$

$$\boldsymbol{I} = \boldsymbol{Z}_0{}^{-1}(s)\left[\exp(-\boldsymbol{Q}(s)x)\boldsymbol{A}(s) - \exp(\boldsymbol{Q}(s)x)\boldsymbol{B}(s)\right]. \tag{6.16}$$

It is however difficult to compute $\boldsymbol{Z}_0$ directly by solving (6.9), and thus we take notice of the following relation:

$$\boldsymbol{Q}^2(s) = (\boldsymbol{Z}_0(s)\boldsymbol{Y}(s))^2 = \boldsymbol{Z}_0(s)\boldsymbol{Y}(s)\boldsymbol{Z}_0(s)\boldsymbol{Y}(s) = \boldsymbol{Z}(s)\boldsymbol{Y}(s). \tag{6.17}$$

This identity allows us to compute $\boldsymbol{Q}(s)$ and $\boldsymbol{Z}_0(s)$ by

$$\boldsymbol{Q}(s) = \sqrt{\boldsymbol{Z}(s)\boldsymbol{Y}(s)}, \tag{6.18}$$

$$\boldsymbol{Z}_0(s) = \boldsymbol{Q}(s)\boldsymbol{Y}^{-1}(s). \tag{6.19}$$

The matrix $\boldsymbol{Q}(s)$ is the propagation constant matrix whose eigenvalues give mode velocities. Although the impedance matrix $\boldsymbol{Z}(s)$ and the admittance matrix $\boldsymbol{Y}(s)$ are symmetric since so are $\boldsymbol{L}, \boldsymbol{C}, \boldsymbol{R}, \boldsymbol{G}$, the propagation constant matrix $\boldsymbol{Q}(s)$ is not necessarily symmetric. The matrix $\boldsymbol{Z}_0(s)$ introduced so as to satisfy Eq. (6.9) is the characteristic impedance matrix and symmetric. Here, let us see the symmetry of $\boldsymbol{Z}_0(s)$. From Eq. (6.18), $\boldsymbol{Q}(s)$ can be represented by a certain polynomial $f(\lambda)$ as [22]

$$\boldsymbol{Q}(s) = \sqrt{\boldsymbol{Z}(s)\boldsymbol{Y}(s)} = f(\boldsymbol{Z}(s)\boldsymbol{Y}(s)). \tag{6.20}$$

Hence, we have

$$\boldsymbol{Y}(s)\boldsymbol{Q}(s)\boldsymbol{Y}^{-1}(s) = \boldsymbol{Y}(s)f(\boldsymbol{Z}(s)\boldsymbol{Y}(s))\boldsymbol{Y}^{-1}(s) = f(\boldsymbol{Y}(s)\boldsymbol{Z}(s)) = f(\boldsymbol{Y}^T(s)\boldsymbol{Z}^T(s)) = \boldsymbol{Q}^T(s), \tag{6.21}$$

$$\therefore\ \boldsymbol{Z}_0(s) = \boldsymbol{Q}(s)\boldsymbol{Y}^{-1}(s) = \boldsymbol{Y}^{-1}(s)\boldsymbol{Q}^T(s) = \boldsymbol{Z}_0{}^T(s). \tag{6.22}$$

6.3 Ground Loss Effects

If a ground has finite permittivity ε_g and finite conductivity σ_g, frequency dependent losses originating from the skin effect of the ground are involved in the impedance matrix $\boldsymbol{Z}(s)$ and the admittance matrix $\boldsymbol{Y}(s)$. The expression for the loss terms was first derived by Carson [23], and later by Sunde [24] and Kikuchi [25]. The Carson's expression is a simplified form of Sunde's one, and the Sunde's is still an approximation of Kikuchi's expression. Those expressions give different results at high frequencies over 100 kHz especially for small ε_g, σ_g such as a dry ground. Hence, we employ Kikuchi's expression hereinafter in this dissertation. Although the paper [25] presents the expression for only a single-phase transmission line, extension to a multi-phase transmission line is straightforward and given by

$$\boldsymbol{Z}(s) = s\boldsymbol{L} + \boldsymbol{R} + \boldsymbol{Z}_g(s) \tag{6.23}$$

$$\boldsymbol{Y}^{-1}(s) = s^{-1}\boldsymbol{C}^{-1} + \boldsymbol{Y}_g^{-1}(s) \tag{6.24}$$

$$\left[\boldsymbol{Z}_g(s)\right]_{ij} = \frac{s\mu_0}{\pi}\int_0^\infty \frac{\mathrm{e}^{-(h_i+h_j)x}\cos(d_{ij}x)}{\sqrt{x^2+\gamma_g^2(s)-\gamma_0^2(s)}+x}\mathrm{d}x \tag{6.25}$$

$$\left[\boldsymbol{Y}_g^{-1}(s)\right]_{ij} = \frac{1}{s\pi\varepsilon_0}\int_0^\infty \frac{\mathrm{e}^{-(h_i+h_j)x}\cos(d_{ij}x)}{\sqrt{x^2+\gamma_g^2(s)-\gamma_0^2(s)}+x\gamma_g^2(s)/\gamma_0^2(s)}\mathrm{d}x \tag{6.26}$$

$$\gamma_g(s) = \sqrt{s\mu_0(\sigma_g+s\epsilon_g)}, \quad \gamma_0(s) = s\sqrt{\mu_0\varepsilon_0}. \tag{6.27}$$

In numerical simulations, we evaluate the integrals of $\boldsymbol{Z}_g(s)$, $\boldsymbol{Y}_g^{-1}(s)$ by the Gauss-Laguerre quadrature of 64 points.

6.4 Numerical Computations

In the numerical simulations, we have to compute $\boldsymbol{Q}(s) = \sqrt{\boldsymbol{Z}(s)\boldsymbol{Y}(s)}$ and $\exp(-\boldsymbol{Q}(s)x)$, namely the square root and the exponential of a matrix. This section presents methods for their computation and a technique of avoiding round-off errors in computation of $\exp(\boldsymbol{Q}(s)x)$ when the frequency dependent losses are incurred.

6.4.1 Computation of Square Root of a Matrix

Given a scalar number a, the Newton's iteration is widely known to compute its square root:

$$x_{k+1} = \frac{1}{2}\left(x_k + \frac{a}{x_k}\right), \tag{6.28}$$

where the initial value x_0 is an arbitrary number other than zero. This algorithm can be extended to its matrix form in a straightforward manner [26] as

$$\boldsymbol{X}_{k+1} = \frac{1}{2}\left(\boldsymbol{X}_k + \boldsymbol{X}_k^{-1}\boldsymbol{A}\right), \quad \boldsymbol{X}_0 = d\boldsymbol{1}, \tag{6.29}$$

where $\boldsymbol{X}_k$ and $\boldsymbol{A}$ are $n\times n$ matrices, $\boldsymbol{1}$ is the $n\times n$ identity matrix, and d is an arbitrary number other than zero. This simple extension is justified as follows. Suppose the matrix $\boldsymbol{A}$ can be diagonalized by an $n\times n$ matrix $\boldsymbol{P}$ as

$$\boldsymbol{P}^{-1}\boldsymbol{A}\boldsymbol{P} = \mathrm{diag}\{\lambda_1, \lambda_2, \cdots, \lambda_n\}, \tag{6.30}$$

where $\{\lambda_i | i = 1, 2, \cdots, n\}$ is the set of eigenvalues of $\boldsymbol{A}$. Then, from Eq. (6.29), it is shown by induction that $\boldsymbol{X}_k$ can also be diagonalized by $\boldsymbol{P}$ as

$$\boldsymbol{P}^{-1}\boldsymbol{X}_k\boldsymbol{P} = \mathrm{diag}\{\lambda_1^{(k)}, \lambda_2^{(k)}, \cdots, \lambda_n^{(k)}\}. \tag{6.31}$$

Therefore, by multiplying Eq.(6.29) by $\boldsymbol{P}^{-1}$ on the left and by $\boldsymbol{P}$ on the right, it reduces to

$$\lambda_i^{(k+1)} = \frac{1}{2}\left(\lambda_i^{(k)} + \frac{\lambda_i}{\lambda_i^{(k)}}\right), \quad \lambda_i^{(0)} = d. \tag{6.32}$$

Since the above iteration coincides with Eq. (6.28), the iteration Eq. (6.29) converges to $\sqrt{\boldsymbol{A}}$.

For a lossless transmission line, Eqs. (6.3) and (6.4) lead to

$$\boldsymbol{Q}(s) = \sqrt{\boldsymbol{Z}(s)\boldsymbol{Y}(s)} = \sqrt{s^2\boldsymbol{L}\boldsymbol{C}} = \frac{s}{c}\boldsymbol{1}, \tag{6.33}$$

where $c = 1/\sqrt{\varepsilon_0\mu_0}$ denotes the speed of light, and thereby $\boldsymbol{Q}^2(s) = \boldsymbol{Z}(s)\boldsymbol{Y}(s)$ is basically a diagonal dominant matrix even if the transmission line is lossy. By taking $d = \sqrt{[\boldsymbol{Z}(s)\boldsymbol{Y}(s)]_{00}}$, we compute $\boldsymbol{Q}(s)$ closest to $(s/c)\boldsymbol{1}$ out of possible 2^n solutions of $\boldsymbol{Q}^2(s) = \boldsymbol{Z}(s)\boldsymbol{Y}(s)$.

6.4.2 Computation of Exponential of a Matrix

In order to compute the exponential of a $n \times n$ matrix $\boldsymbol{A} = (a_{ij})$, we employ the scaling and squaring method [27]. With the method, the matrix $\boldsymbol{A}$ is first scaled down to $2^{-k}\boldsymbol{A}$, where k is the minimum integer such that

$$n \max_{i,j} |a_{ij}| < 2^k. \tag{6.34}$$

Then, since the following inequality holds for an arbitrary vector $\boldsymbol{x}$

$$\|\boldsymbol{A}\boldsymbol{x}\| \le n \max_{i,\,j} |a_{ij}| \|\boldsymbol{x}\|, \tag{6.35}$$

we have

$$\|\boldsymbol{A}\| \equiv \sup_{\|\boldsymbol{x}\|=1} \|\boldsymbol{A}\boldsymbol{x}\| \le n \max_{i,\,j} |a_{ij}| < 2^k, \tag{6.36}$$

$$\therefore \ \|2^{-k}\boldsymbol{A}\| < 1. \tag{6.37}$$

Hence, the modulus of every eigenvalue of $2^{-k}\boldsymbol{A}$ is assured to be less than 1. The property avoids numerical errors in computation of $\exp(2^{-k}\boldsymbol{A})$ by using the Taylor expansion

$$\exp(\boldsymbol{X}) = \sum_{i=0}^{\infty} \frac{\boldsymbol{X}^i}{i!} = \boldsymbol{1} + \frac{\boldsymbol{X}}{1!} + \frac{\boldsymbol{X}^2}{2!} + \cdots, \tag{6.38}$$

since the series becomes monotonically decreasing in modulus so that we do not have to take care of the truncation point considering eigenvalues of $\boldsymbol{A}$. Note that the convergence is much

different when an eigenvalue of $\boldsymbol{X}$ is, say, -1000. After that, by squaring $\exp(2^{-k}\boldsymbol{A})$ for k times, we have $\exp(\boldsymbol{A})$ as

$$\exp(\boldsymbol{A}) = \left[\exp(2^{-k}\boldsymbol{A})\right]^{2^k} = \left[\exp(2^{-k}\boldsymbol{A})\right]^{\overbrace{2\cdots 2}^{k}}. \tag{6.39}$$

6.4.3 Avoiding Divergence of $\exp(\boldsymbol{Q}(s)x)$

Let us consider to solve Eqs. (6.15) and (6.16) numerically for a transmission line of length ℓ. With some boundary conditions at $x = 0, \ell$, we compute the exponential of $\pm\boldsymbol{Q}(s)x$ for $x = 0, \ell$ and obtain the coefficients $\boldsymbol{A}(s), \boldsymbol{B}(s)$ as the solution. However, we had some unexpected diverging oscillations in the resulting transient waveforms for long three-transmission lines above a lossy ground.

One of the cases was for a 30 km-long three-transmission line above a dry ground ($\varepsilon_g = 4\varepsilon_0, \sigma_g = 10^{-4}$ S/m). The eigenvalues and eigenvectors of $\boldsymbol{Q}(s)$ for a value of $s = 3500 + \mathrm{j}6.43 \times 10^6\ \mathrm{s}^{-1}$ were

$$\begin{aligned}
\lambda_1 &= 1.382 + \mathrm{j}22.16\ \mathrm{km}^{-1}, \quad \boldsymbol{x}_1 = (1, 1.007 + \mathrm{j}0.007, 1),\\
\lambda_2 &= 0.026 + \mathrm{j}21.47\ \mathrm{km}^{-1}, \quad \boldsymbol{x}_2 = (1, 0, -1),\\
\lambda_3 &= 0.011 + \mathrm{j}21.44\ \mathrm{km}^{-1}, \quad \boldsymbol{x}_3 = (1, -2.174 + \mathrm{j}0.031, 1).
\end{aligned}$$

Since $\boldsymbol{x}_1 \simeq (1, 1, 1)$, the λ_1 corresponds to the earth return mode which is the most attenuated one by the loss of the ground. The remaining λ_2, λ_3 correspond to aerial modes since the sum of all elements of their eigenvector is nearly equal to zero. The aerial modes propagate along the conductors, and are less affected by the loss of the ground. For $x = \ell = 30$ km, the attenuation is given by

$$|\exp(-\lambda_1\ell)| = 0.994 \times 10^{-18}, \quad |\exp(-\lambda_2\ell)| = 0.450, \quad |\exp(-\lambda_3\ell)| = 0.700. \tag{6.40}$$

The amplitude difference of 10^{-18} exceeds the precision of `double`, the 64-bit floating point data type. Consequently, the λ_1 mode, which is physically attenuated with propagation, numerically disappears in the computation of $\exp(-\boldsymbol{Q}(s)\ell)$ due to the round-off errors. On the contrary, the λ_2, λ_3 modes, which are the principal propagation modes, numerically disappear in the computation of $\exp(\boldsymbol{Q}(s)\ell)$ due to the round-off errors while the physically attenuated λ_1 mode remains through the computation. Thus, the computation of $\exp(\boldsymbol{Q}(s)\ell)$ results in physically meaningless values, and causes the unexpected oscillations.

To avoid the computation of $\exp(\boldsymbol{Q}(s)\ell)$, we alter the definition of $\boldsymbol{B}(s)$ as follows:

$$\boldsymbol{V} = \exp(-\boldsymbol{Q}(s)x)\boldsymbol{A}(s) + \exp(-\boldsymbol{Q}(s)(\ell - x))\boldsymbol{B}(s), \tag{6.41}$$

$$\boldsymbol{I} = \boldsymbol{Z}_0{}^{-1}(s)\left[\exp(-\boldsymbol{Q}(s)x)\boldsymbol{A}(s) - \exp(-\boldsymbol{Q}(s)(\ell - x))\boldsymbol{B}(s)\right]. \tag{6.42}$$

From a physical viewpoint, $\boldsymbol{A}(s)$ represents the incident voltage at $x = 0$, and the redefined $\boldsymbol{B}(s)$ represents the incident voltage at $x = \ell$. The above set of equations is symmetric with respect to change of the variable x with $y = \ell - x$, where the new variable y denotes the distance from the other end of the line. The symmetry implies that the redefined set of equations corresponds to the notion of the scattering matrix of transmission lines.

Chapter 7

Expression of Fault Transients by Laguerre Functions

7.1 Introduction

Laguerre functions form an orthonormal basis, and their asymptotic exponential form is exploited especially for transient applications such as circuit synthesis [28], filter design [29, 30], system identification [31], and numerical inversion of Laplace transforms [4]. While those applications are based on the Laguerre expansion of a function, we show in this chapter that the Laguerre functions directly appear as impulsive waveforms at a fault occurrence on transmission lines.

The transient phenomena at faults on transmission lines have been discussed from the viewpoint of protection of power systems [32, 33, 34, 35, 36]. However, transient waveforms have not been investigated analytically. We derive analytical expressions for the transient waveforms assuming ideal single-phase transmission lines, and show that they are identical to the Laguerre functions. Moreover, it is shown that the remarkable property of the transient waveforms applies to faults on ungrounded multi-phase transmission lines with some simplifications.

As an application of the expressions, we propose a novel method of fault location. Traveling-wave-based location methods observe transient waveforms to find impulsive waveforms that appear periodically due to multiple reflections of the surge. To find them effectively, the correlation technique [37, 38, 39] and the use of the wavelets [40, 41] have been proposed thus far. Our method utilizes the orthogonality of the impulsive waveforms that are found to be the Laguerre functions, and gives the possibility of a more accurate method. Further, we evaluate effects of a lossy ground with which transients are no longer the Laguerre functions.

In this chapter, the proposed numerical inversion of Laplace transforms is used to compute transient waveforms, which is indispensable especially when evaluating ground loss effects since the loss terms are difficult to invert analytically.

7.2 Transient Waveforms at Faults

We consider transient phenomena at a fault on a lossless single-phase transmission line modeled as shown in Fig. 7.1. The fault occurs at position $x = d$ and at time $t = 0$, and consequently the circuit is shorted at $x = d$ for $t \geq 0$. By the principle of superposition, the short circuit may be analyzed with individual circuits shown in Fig. 7.2, in which the voltage source $\tilde{e}(t)$ is arbitrary. The voltage across the sending end, $v_S(t)$, is given by the superposition as

$$v_S(t) = v_{Sa}(t) + v_{Sb}(t). \tag{7.1}$$

Let $v_{nf}(x, t)$ be the voltage on the transmission line if the fault does not occur, and let

$$\tilde{e}(t) = v_{nf}(d, t). \tag{7.2}$$

Then, the behavior of the circuit (a) in Fig. (7.2) is nothing but that of the original circuit shown in Fig. (7.1) if the fault does not occur. In that case, $v_{Sa}(t)$ represents the steady state component of $v_S(t)$ given by

$$v_{Sa}(t) = v_{nf}(0, t). \tag{7.3}$$

As for $v_{Sb}(t)$, it represents the transient component caused by the fault.

Figure 7.3 presents $v_{Sb}(t)$ computed by the numerical Laplace inversion proposed in Chapter 3. Its Laplace transform is obtained by Eq. (6.41) and boundary conditions with parameters

$$L = 1.80\,\text{mH/km}, \quad C = 0.00618\,\mu\text{F/km},$$

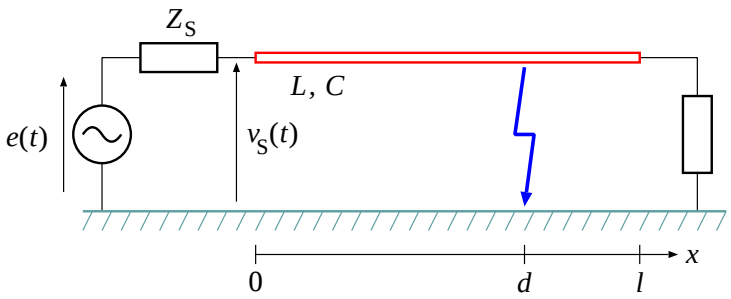


Figure 7.1: The model of a fault on a single-phase transmission line.

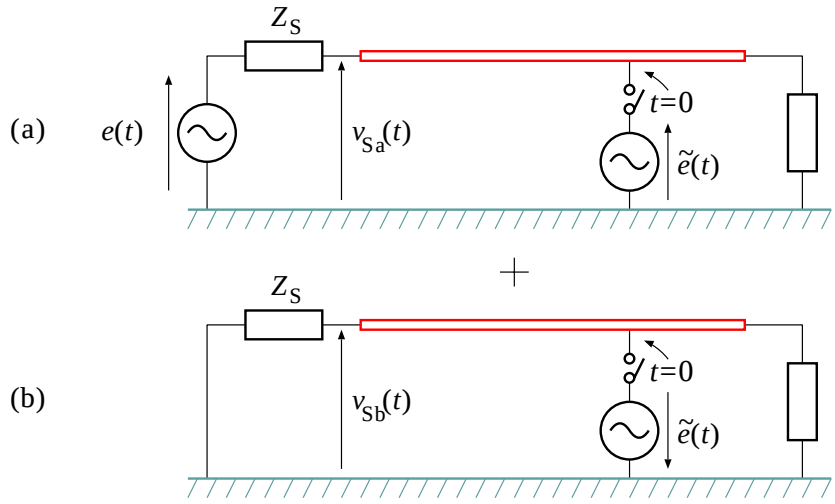


Figure 7.2: The superposition of the short circuit.

$$Z_S(s) = sL_S, \quad L_S = 1.4\,\text{mH}, \quad d = 30\,\text{km},$$
$$e(t) = E_0 \cos(2\pi f t + \theta), \quad f = 60\,\text{Hz},$$

where we assume a lossless conductor at height 10 m and of radius 2.5 mm above a perfectly conducting ground. We neglected the resistance part of $Z_S(s)$ since the transient waveform is composed of high frequency components as shown in Fig. 7.3. For those high frequencies, the inductance part, sL_S, dominates over the resistance part. In addition, the steady state voltage at the fault point, $v_{nf}(d,t)$, can be expressed as

$$v_{nf}(d,t) = E\cos(2\pi f t + \phi), \tag{7.4}$$

where E denotes the amplitude and ϕ is the initial phase. In the above example, we assume $\phi = \pi$, that is, we assume that the line voltage at the fault point reached its (negative) maxima at the fault occurrence ($t = 0$). Then, the surge waveform $-\tilde{e}(t)$ in Fig. 7.2 (b) is given by

$$-\tilde{e}(t) = -v_{nf}(d,t) = E\cos(2\pi f t). \tag{7.5}$$

This waveform is much like the step waveform in the time range sufficiently shorter than the period $1/f = 16.7$ ms. Hence, it can be thought of as the step waveform when analyzing the transient waveform for $0 \le t \le 1$ ms as shown in Fig. 7.3.

In Fig. 7.3, impulsive waveforms appear periodically at the sending end due to multiple reflections of the surge induced by the fault. Right after the fault, the step-like surge propagates

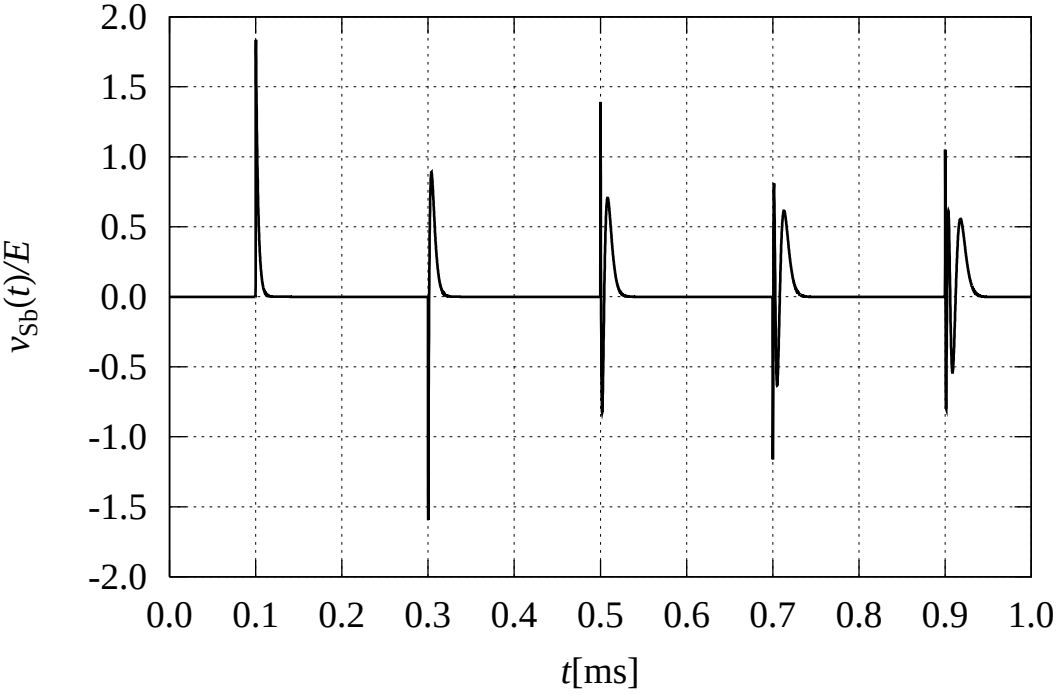


Figure 7.3: Normalized transient waveform $v_{Sb}(t)/E$ when a fault occurred at $d = 30\,\text{km}, t = 0\,\text{ms}$. ($N = 8192, T = 0.0005, aT = 3.5, i = 3$)

along the line and arrives at the sending end for the first time. The first impulsive waveform at $t = 0.1\,\text{ms}$ in Fig. 7.3 originates from this first arrival of the surge. To be specific, the waveform is an exponential function formed by the step-like surge applied to the sending end inductance L_S with the characteristic impedance $Z_0(s)$. Having been reflected at the sending end, the surge propagates back to the fault point and is reflected there at $t = 0.2\,\text{ms}$. When it reaches the sending end for the second time, the second impulsive waveform at $t = 0.3\,\text{ms}$ in Fig. 7.3 appears. A similar consideration leads to the understanding that the impulsive waveforms stem from the multiple reflections of the surge which goes back and forth between the sending end and the fault point. Hence, the time interval in which the impulsive waveforms appear is equal to the time for the surge to take one trip. In the above case, the propagation velocity on the transmission line is given by

$$c = \frac{1}{\sqrt{LC}} = 3.00 \times 10^8\,\text{m/s}, \tag{7.6}$$

and then the time interval is

$$\frac{2d}{c} = \frac{2 \cdot 30\,\text{km}}{3.00\,\text{m/s}} = 0.20\,\text{ms}. \tag{7.7}$$

This value coincides with the time interval seen from Fig. 7.3.

7.3 Analytical Expressions for Transient Waveforms

In this section, we derive analytical expressions for the impulsive waveforms with some simplifications. We neglect the resistance part of the sending end impedance $Z_S(s)$ as in the previous section, and in the short time range, we approximate the surge waveform by the constant as

$$-\tilde{e}(t) = E. \tag{7.8}$$

Figure 7.4 illustrates the simplified circuit of Fig. 7.2 (b), in which only the part relevant to $v_{Sb}(t)$ is illustrated. Moreover, since the transmission line is lossless, the propagation constant and the characteristic impedance of the line are given by

$$Q(s) = s\sqrt{LC} = \frac{s}{c}, \quad Z_0(s) = Z_0 = \sqrt{\frac{L}{C}}, \tag{7.9}$$

and thereby the line can simply be treated as a delay element of delay time $T_d = d/c$.

Now, let us denote the normalized impulsive waveforms shown in Fig. 7.3 from the left by $u_0(t)$, $u_1(t)$, $\cdots$, and their Laplace transforms by $U_0(s)$, $U_1(s)$, $\cdots$, such that the origin of each $u_j(t)$ is shifted to the time point when the impulsive waveform appears. Then, $v_{Sb}(t)$ can be expressed by the sum of shifted $u_j(t)$ as follows:

$$v_{Sb}(t) = E\sum_{j=0}^{\infty} u_j(t - T_d(1 + 2j)). \tag{7.10}$$

When the step surge arrives at the sending end for the first time, $u_0(t)$ is observed as the transient waveform. Hence, as the expression of $U_0(s)$, we have

$$U_0(s) = \frac{2Z_S(s)}{Z_S(s) + Z_0(s)}\frac{1}{s} = \frac{2}{s + \alpha}, \tag{7.11}$$

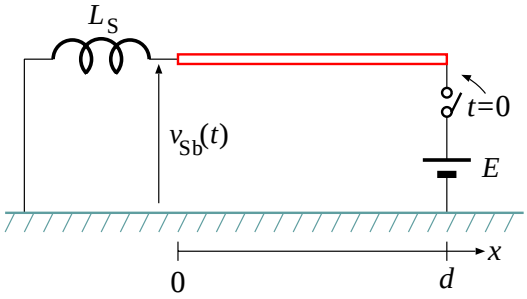


Figure 7.4: The simplified circuit of Fig. 7.2 (b).

where $\alpha = Z_0/L_S$. The term $1/s$ is the Laplace transform of the normalized surge waveform $-\tilde{e}(t)/E = 1$. Having been reflected at the sending end and at the fault point in turn, it arrives at the sending end for the second time. As the transient waveform, $u_1(t)$ is observed this time, and then we have

$$U_1(s) = \frac{2Z_S(s)}{Z_0(s) + Z_S(s)} \cdot (-1) \cdot \frac{Z_S(s) - Z_0(s)}{Z_S(s) + Z_0(s)} \frac{1}{s} = -\frac{2}{s+\alpha} \frac{s-\alpha}{s+\alpha}. \tag{7.12}$$

A similar consideration leads to the following expression for the j-th impulsive waveform

$$U_j(s) = \frac{2}{s+\alpha} \left(-\frac{s-\alpha}{s+\alpha} \right)^j. \tag{7.13}$$

The analytical Laplace inversion of $U_j(s)$ is given by

$$u_j(t) = 2(-1)^j \mathrm{e}^{-\alpha t} L_j(2\alpha t), \tag{7.14}$$

$$L_j(t) = \sum_{k=0}^{j} \frac{{}_jC_k}{k!} (-t)^k, \tag{7.15}$$

where $L_j(t)$ is the j-th Laguerre polynomial, and ${}_nC_k$ denotes the binomial coefficient. Thus, the impulsive waveform $u_j(t)$ is identical to the j-th Laguerre function except for the time scaling of 2α and the scalar multiple. It follows immediately that $u_j(t)$ are orthogonal to each other as

$$\int_0^\infty u_i(t) u_j(t)\, \mathrm{d}t = \frac{2}{\alpha} \delta_{ij}, \tag{7.16}$$

where δ_{ij} is the Kronecker's delta.

In the above derivations, the fault phase is assumed to be $\phi = \pi$ in Eq. (7.4), and therefore the surge waveform can be approximated by the step waveform. If a fault occurs at phase $\phi = \pi/2$, namely, if a fault occurs when the line voltage at the fault point is zero, the approximation is no longer applicable. In that case, the surge waveform $-\tilde{e}(t)$ and the transient waveform $v_{Sb}(t)$ are unobservably small. Indeed, they become about one-hundredth in amplitudes with the parameters considered in the previous section. In general case, the surge waveform is given by

$$-\tilde{e}(t) = -v_{nf}(d, t) = -E \cos\phi \cdot \cos(2\pi f t) + E \sin\phi \cdot \sin(2\pi f t). \tag{7.17}$$

This is a linear combination of the two kinds of surge waveforms corresponding to $\phi = \pi$ and $\phi = \pi/2$. Since transients caused by the latter term is negligibly small, the transient waveform $v_{Sb}(t)$ can be expressed approximately by

$$v_{Sb}(t) \simeq E \cos\phi \sum_{j=0}^{\infty} u_j(t - T_d(1 + 2j)). \tag{7.18}$$

7.4 Extension to Three-Phase Transmission Lines

In the actual power systems, three-phase transmission lines are mainly used, and therefore we are motivated to extend the previous result to ungrounded three-phase transmission lines shown in Fig. 7.5. However, transient waveforms observed at faults on the three-phase transmission lines are much more complicated than those observed on the single-phase transmission lines since multiple reflections of surges at three sending/receiving end terminals and the fault point are involved as seen from Fig. 7.5. We investigate transients at a single-phase-to-ground fault and at a phase-to-phase fault, which are the most possible faults, and show that transient waveforms are also expressed by the Laguerre functions with some simplifications. The extension actually turns out to apply to other ungrounded multi-phase transmission lines.

7.4.1 Single-Phase-to-Ground Faults

In Fig. 7.5, the single-phase-to-ground fault is labeled (i). The circuit is shorted between the conductor #1 and the ground at position $x = d$, and at time $t = 0$. The voltage sources $e_1(t)$, $e_2(t)$, $e_3(t)$ form a symmetric three-phase voltage source. If the fault does not occur, the voltage of phase #1 at $x = d$ in the steady state is given similarly as Eq. (7.4) by

$$v_{1\mathrm{nf}}(d,t) = E_1 \cos(2\pi f t + \phi_1). \tag{7.19}$$

As for the fault phase, we again assume that $\phi_1 = \pi$, namely that the voltage of phase #1 at the

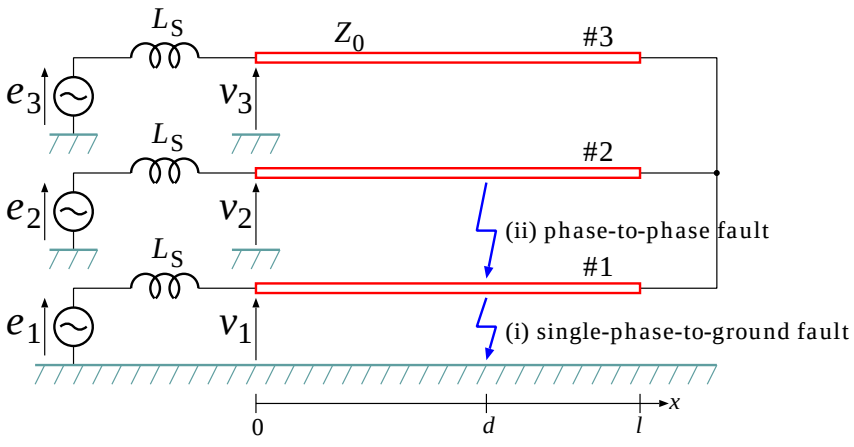


Figure 7.5: Considered types of faults on three-phase transmission lines.

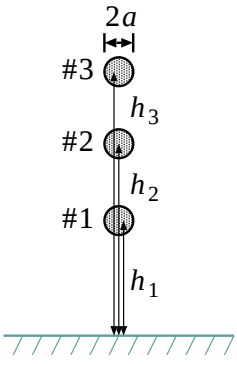


Figure 7.6: Configuration of conductors.

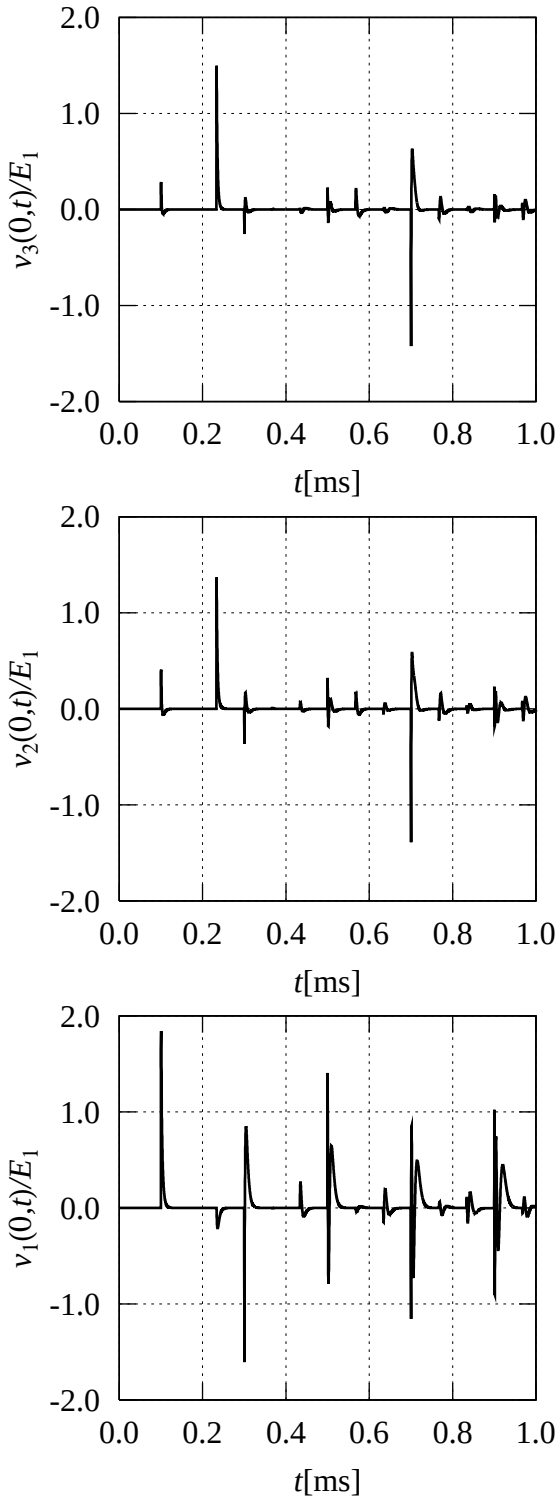


Figure 7.7: Normalized transient waveforms at a single-phase-to-ground fault. ($N = 8192, T = 0.0005, aT = 3.5, i = 3$)

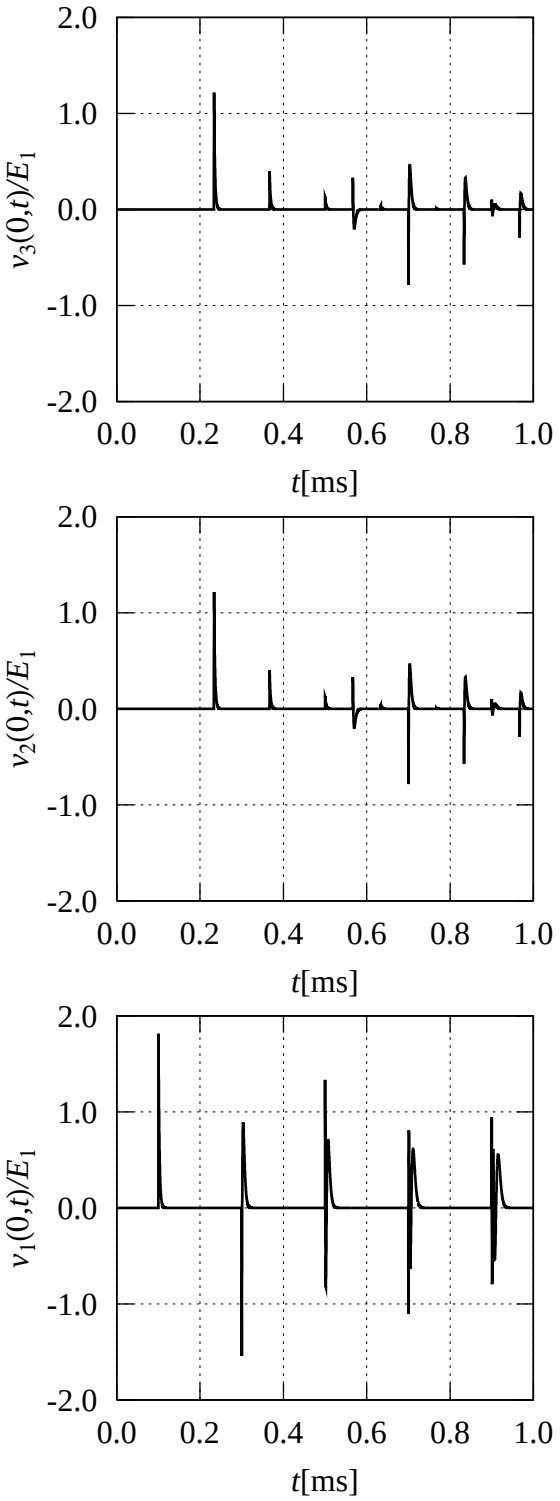


Figure 7.8: Normalized transient waveforms with the simplifications. ($N = 8192, T = 0.0005, aT = 3.5, i = 3$)

fault point reached its (negative) maxima, $v_1(d,0) = -E_1$, at the fault occurrence. Figure 7.7 shows transient voltage waveforms at the sending end, in which the line length and the fault point are $\ell = 50\,\mathrm{km}, d = 30\,\mathrm{km}$, respectively, and the conductors are configured as shown in Fig. 7.6 with $h_1 = 30\,\mathrm{m}, h_2 = 40\,\mathrm{m}, h_3 = 50\,\mathrm{m}, a = 20\,\mathrm{mm}$.

On the lossless transmission lines, the propagation constant matrix $\boldsymbol{Q}(s)$ becomes diagonal as Eq. (6.33). Then, the voltages and the currents on conductors do not interfere with each other when they are propagating along the line. The characteristic impedance matrix $\boldsymbol{Z}_0(s)$, however, is not diagonal, and thereby when a fault occurs on the phase #1, the surge is induced not only on the phase #1 but also on the other phases #2 and #3. Moreover, part of voltage/current on a phase conductor is reflected into the other phases at end points including the fault point. From a physical viewpoint, those phenomena are caused by the capacitive and the inductive coupling of the phase conductors. For example, at $t = 0.1\,\mathrm{ms}$ in Fig. 7.7, one can see the first impulsive waveform in $v_1(0,t)$ caused by the surge, and at the same time, small spikes in $v_2(0,t), v_3(0,t)$. The spikes stem from the surges induced on the phases #2 and #3 at the fault occurrence, and the reflection at the sending end. The off-diagonal elements of $\boldsymbol{Z}_0(s)$ cause those phenomena, though they are rather small. Hence we neglect them for simplicity in the following derivation. In that case, the three-phase transmission line becomes merely a set of three independent single-phase transmission lines. Figure 7.8 shows the transient waveforms computed for the circuit labeled (i) in Fig. 7.9 with the simplification. In the following, we derive analytical expressions for them.

Let us begin with $v_1(0,t)$ which looks the same as the transient waveform on single-phase

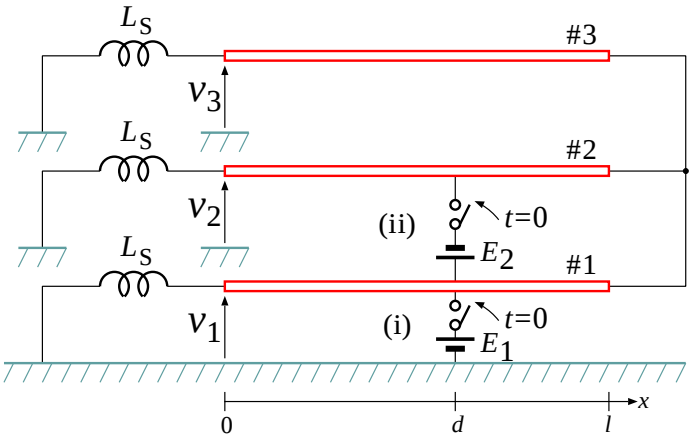


Figure 7.9: The corresponding circuits to the transient waveforms.

transmission lines. In fact, the left part from the fault point of the conductor #1 forms the same circuit with the single-phase case shown in Fig. 7.4. Hence, $v_1(0,t)$ is expressed by Eq. (7.10) as

$$v_1(0,t) = E_1 \sum_{j=0}^{\infty} u_j(t - T_d(1+2j)). \tag{7.20}$$

To derive expressions for $v_2(0,t)$ and $v_3(0,t)$, we further assume that $h_1 = h_2 = h_3 = 40\,\mathrm{m}$. The assumption leads to $\boldsymbol{Z}_0 = Z_0\boldsymbol{I}, Z_0 = \sqrt{L/C}$ and thereby all the propagation constants become the same. Note that this assumption does not necessarily mean that the three conductors are configured at the same place since we already simplified $\boldsymbol{Z}_0(s)$ to be diagonal and the spatial relationship is no longer considered. Moreover, $v_2(0,t)$ and $v_3(0,t)$ are identical to each other from a symmetry of the circuit, and thereby the conductors #2 and #3 may be combined into one conductor for the sake of analysis as illustrated in Fig. 7.10. The characteristic impedance of the combined conductor #2+#3 and the combined sending end inductance are equal to $Z_0/2, L_S/2$, respectively. The voltage reflection/transmission coefficients at the end points are shown in Fig. 7.10.

When the surge arrives at the sending end in the course of the multiple reflections, a Laguerre function $u_j(t)$ appears in a similar way as analyzed in the previous section. In Fig. 7.10, first seven expected impulsive waveforms are calculated on the right based on the lattice diagram. We can see their correspondence to the impulsive waveforms shown in Fig. 7.8. Note that the sixth waveform is sum of two waveforms which take distinct paths. It is a common situation that impulsive waveforms originating from multiple distinct paths are observed at the same time if d/l is rational. Let us denote by τ_1 the time the surge takes to travel between the fault point and the receiving end, i.e. $(\ell - d)/c$, and by τ_2 the time to travel between the receiving end and the sending end, i.e. ℓ/c. Then, time points when impulsive waveforms appear at the sending end of the combined conductor #b+#c can be expressed by

$$(2m+1)\tau_1 + (2n+1)\tau_2, \quad m,n = 0,1,2,\cdots, \tag{7.21}$$

where m and n are integers representing the number of reflections, in the paths, at the fault point and at the sending end, respectively, as shown in Fig. 7.11.

In order to derive general expressions for $v_2(0,t), v_3(0,t)$, let r be the voltage reflection coefficient at the receiving end on the side of the conductor #1. In the case of three-phase transmission lines, r is $-1/3$. Then, other voltage reflection and transmission coefficients at the receiving end are represented by r as shown in Fig. 7.11. Let us consider an impulsive waveform, which appears at the sending end of #2+#3 at $t = (2m+1)\tau_1 + (2n+1)\tau_2$. It is composed of multiple

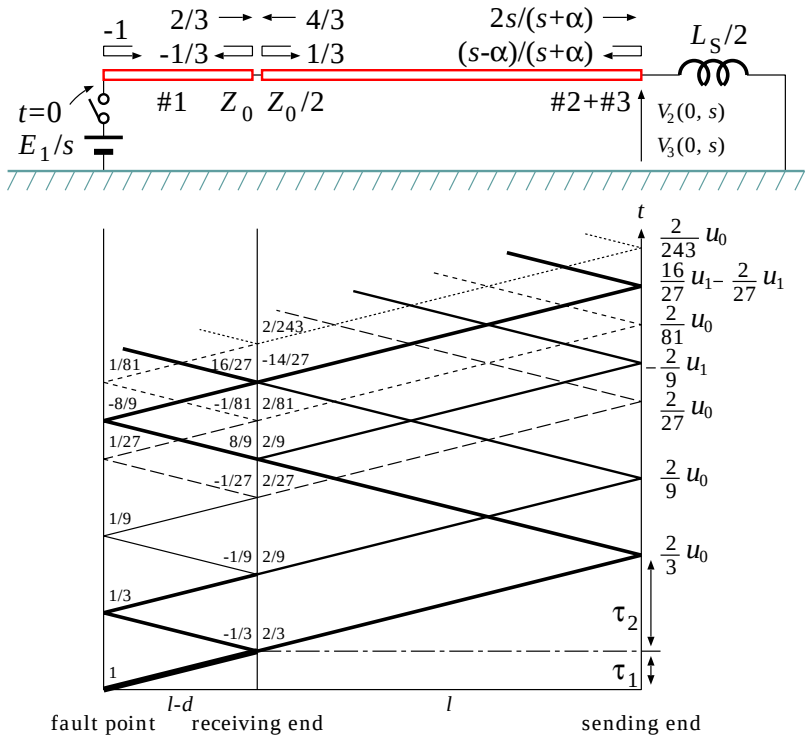


Figure 7.10: The circuit with #2 and #3 combined and its lattice diagram.

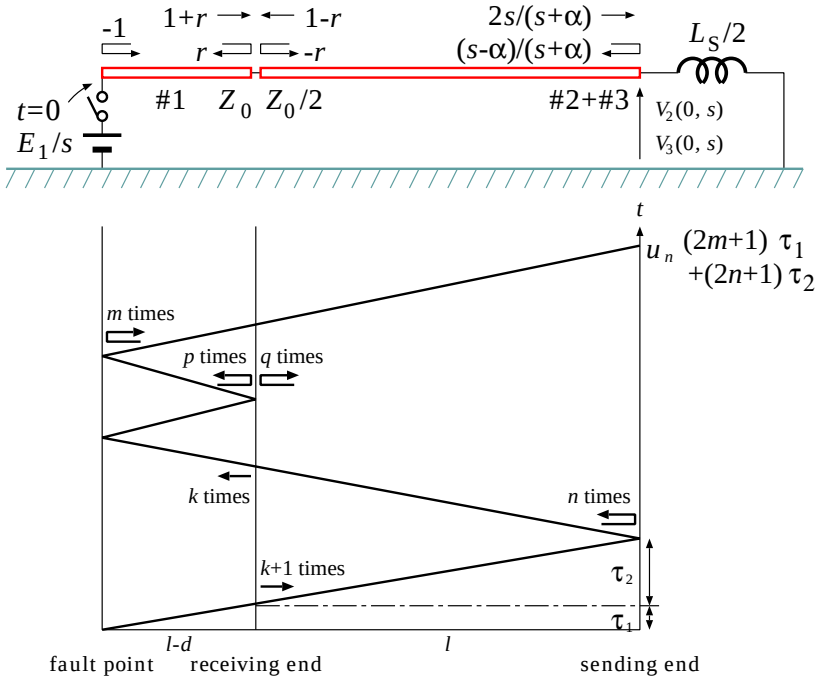


Figure 7.11: One of the paths involved in the transient waveform corresponding to m, n, k.

waveforms that take distinct paths including m times reflections at the fault point and n times reflections at the sending end. Picking up one of those paths, we define another three integers (see Fig. 7.11): k as the number of transmissions at the receiving end from #2+#3 to #1, p as the number of reflections at the receiving end on the side of #1, and q as the number of reflections at the receiving end on the side of #2+#3. Then, the number of transmissions at the receiving end from #1 to #2+#3 is equal to $k + 1$, and the following relations hold:

$$m = p + k, \quad n = q + k. \tag{7.22}$$

Since p, q are equal to or greater than zero, the range of possible k is

$$k = 0,\ 1,\ \cdots,\ \min(m,\ n). \tag{7.23}$$

Thus, the expression for the impulsive waveforms corresponding to k is given by

$$\begin{aligned} &\frac{1}{s}(-1)^m \left(\frac{s-\alpha}{s+\alpha}\right)^n r^p (-r)^q (1+r)^{k+1}(1-r)^k \frac{2s}{s+\alpha} \\ &= \frac{2}{s+\alpha}\left(\frac{s-\alpha}{s+\alpha}\right)^n (-r)^{m+n}(1+r)(-1)^k \left(\frac{1-r^2}{r^2}\right)^k . \end{aligned} \tag{7.24}$$

There might still be multiple paths which correspond to k since there remains an ambiguity of the order of the reflections and transmissions specified by m, n, p, q and k. The number of such paths is given by ${}_mC_k \cdot {}_nC_k$, since we can choose $k + 1$ places out of $m + 1$ where the surge passes over the receiving end from #1 to #2+#3, and independently, k places out of n where the surge passes over the receiving end from #2+#3 to #1. Hence, we have the following analytical expression for the impulsive waveform specified by m and n:

$$W_{m,n}(s) = \frac{2}{s+\alpha}\left(\frac{s-\alpha}{s+\alpha}\right)^n (-r)^{m+n}(1+r) \sum_{k=0}^{\min(m,\,n)} (-1)^k \left(\frac{1-r^2}{r^2}\right)^k {}_mC_k \cdot {}_nC_k. \tag{7.25}$$

From Eqs. (7.13) and (7.14), the Laplace inversion of $W_{m,n}(s)$ is given by

$$w_{m,n}(t) = (-1)^m r^{m+n}(1+r)u_n(t) \sum_{k=0}^{\min(m,\,n)} (-1)^k \left(\frac{1-r^2}{r^2}\right)^k {}_mC_k \cdot {}_nC_k. \tag{7.26}$$

The transient waveform $w_{m,n}(t)$ is equal to $u_n(t)$ multiplied by the constant determined by m, n, and therefore it is also expressed by the Laguerre functions. Finally, we have

$$v_2(0,t) = v_3(0,t) = E \sum_{m,n=0}^{\infty} w_{m,n}(t - (2m+1)\tau_1 - (2n+1)\tau_2). \tag{7.27}$$

Note that the expression Eq. (7.26) can easily be extended to the case of single-phase-to-ground faults on n-phase transmission lines. In that case, the conductor #2+#3 in Fig. 7.10 is replaced by the combined $n-1$ conductors whose characteristic impedance is $Z_0/(n-1)$. Then, by simply taking

$$r = \frac{Z_0/(n-1) - Z_0}{Z_0/(n-1) + Z_0} = -\frac{n-2}{n}, \tag{7.28}$$

the above arguments still hold for n-phase transmission lines.

7.4.2 Phase-to-Phase Faults

In Fig. 7.5, the phase-to-phase fault is labeled (ii). The circuit is shorted between the conductor #1 and #2 at position $x = d$, and at time $t = 0$. If the fault does not occur, the voltage of phase #1 with reference to phase #2 at $x = d$ in the steady state is given by

$$\begin{aligned} v_{12\mathrm{nf}}(d,t) &= E_1\cos(2\pi ft + \phi_1) - E_1\cos(2\pi ft + \phi_1 + 2\pi/3) \\ &= E_2\cos(2\pi ft + \phi_2), \end{aligned} \tag{7.29}$$

$$E_2 = \sqrt{3}E_1, \quad \phi_2 = \phi_1 - \tan^{-1}\frac{1}{\sqrt{3}}. \tag{7.30}$$

We assume $\phi_2 = \pi$ as well in the following.

Figure 7.12 shows transient voltage waveforms at the sending end. The line length and the fault point are the same as in the previous section, i.e. $\ell = 50\,\mathrm{km}, d = 30\,\mathrm{km}$, and the conductors are also configured as shown in Fig. 7.6 with $h_1 = 30\,\mathrm{m}, h_2 = 40\,\mathrm{m}, h_3 = 50\,\mathrm{m}, a = 20\,\mathrm{mm}$. Taking into account the simplifications $h_1 = h_2 = h_3 = 40\,\mathrm{m}$ and $\boldsymbol{Z}_0(s) = Z_0\mathbf{1}$, Fig. 7.13 shows the transient waveforms computed for the circuit labeled (ii) in Fig. 7.9.

The conductors #1 and #2 can be considered to form a single-phase transmission line, and thereby the transient waveforms $v_1(0,t)$ and $v_2(0,t)$ are identical to that on single-phase transmission lines except for their signs. Moreover, since $v_1(x,t) + v_2(x,t) = 0$ holds for any x and t, the surge is not reflected at the right end into the conductor #3. Thus, we have the following expressions for the transient waveforms:

$$v_1(0,t) = -v_2(0,t) = \frac{E_2}{2}\sum_{j=0}^{\infty} u_j(t - T_d(1+2j)) \tag{7.31}$$

$$v_3(0,t) = 0. \tag{7.32}$$

These expressions also apply to the phase-to-phase faults of multi-phase transmission lines.

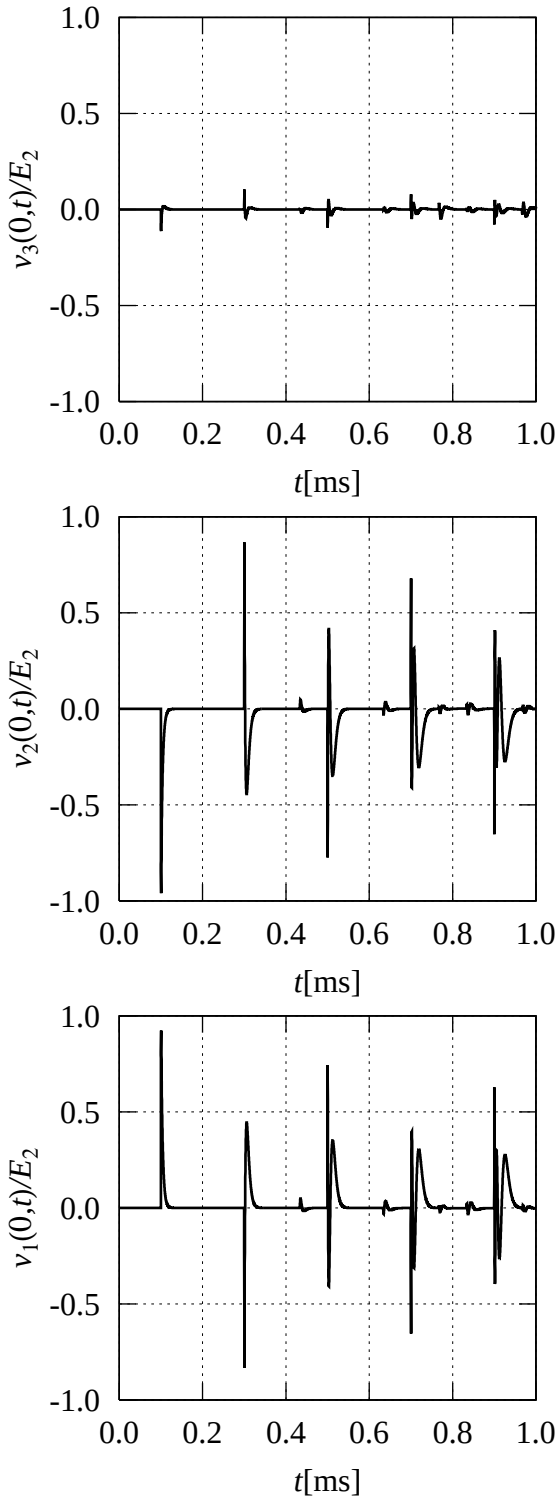


Figure 7.12: Normalized transient waveforms at a phase-to-phase fault. ($N = 8192, T = 0.0005, aT = 3.5, i = 3$)

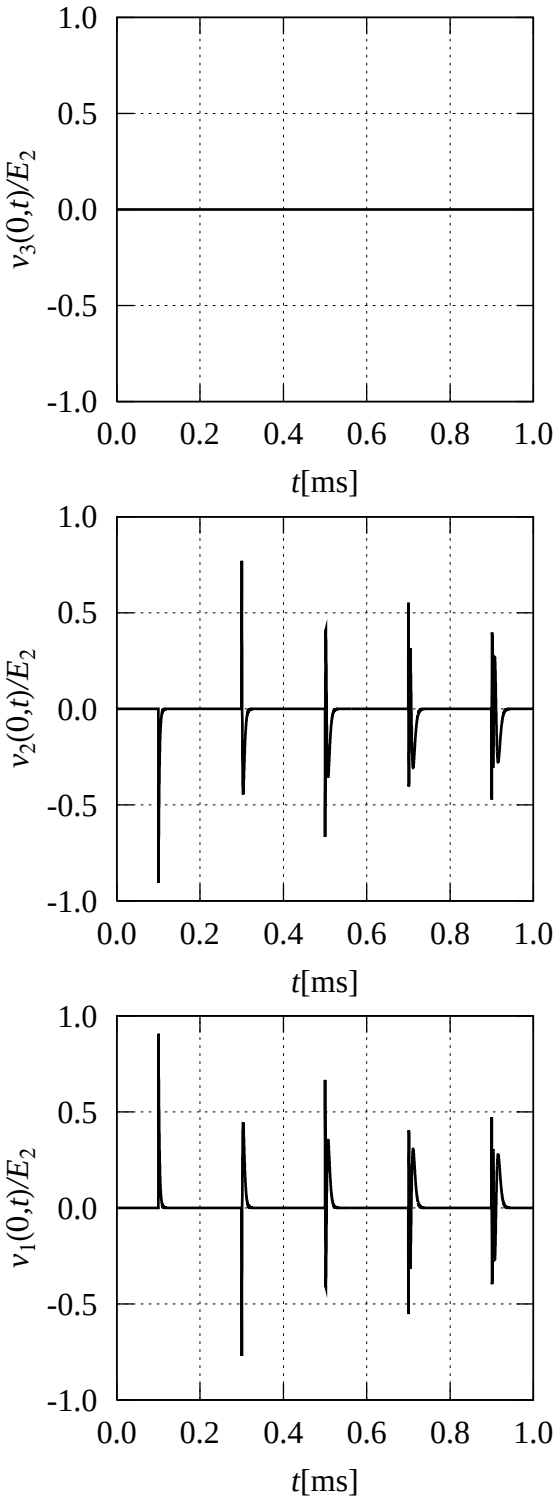


Figure 7.13: Normalized transient waveforms with the simplifications. ($N = 8192, T = 0.0005, aT = 3.5, i = 3$)

7.5 Application to Fault Location

As an application of the expressions for the transient waveforms, we propose a novel fault location method. The traveling-wave-based fault location techniques are based on extraction of the interval at which the impulsive waveforms appear at the sending end. We propose a method to extract the impulsive waveforms by using the orthogonality of the Laguerre functions.

7.5.1 Correlation Filter Bank

Equation (7.14) shows that the impulsive waveforms depend on $\alpha = Z_0/L_S$ which is determined regardless of the fault point so that we can compute the waveforms in advance of a fault occurrence. We construct a set of correlation filters shown in Fig. 7.14. Each of the filters holds a precomputed impulsive waveform and computes correlation between the transient waveform and the impulsive waveform as follows:

$$r_j(n) = \sum_{k=0}^{L-1} v_{Sb}((n+k)T_S)u_j(kT_S), \tag{7.33}$$

where T_S denotes the sampling period and L is the filter length which is sufficiently large so that $u_j(kT_S), k \geq L$ are negligibly small. When an impulsive waveform $u_j(t)$ appears in their input, the output of corresponding correlation filter $r_j(n)$ will reach its maxima, and the output of the other filters will vanish to zero due to the orthogonality. The discriminator in Fig. 7.14 works to find such time points. The following code describes the algorithm we used in numerical simulations for the discriminator:

```
n := 0;
for i := 0 to N − 1 do begin
  while ∃j ≠ i, |r_i(n)| < K1|r_j(n)| or |r_i(n)| < K2
        or |r_i(n)| < |r_i(n + 1)| or |r_i(n + 1)| < |r_i(n + 2)| do begin
    n := n + 1;
  end
  Time[i] := nT_S;
end
```

The integer N denotes the number of filters, and $K_1(> 1)$ is a threshold to discriminate between the output $|r_i(n)|$ which is supposed to reach its maxima and others $|r_j(n)|$ which are supposed

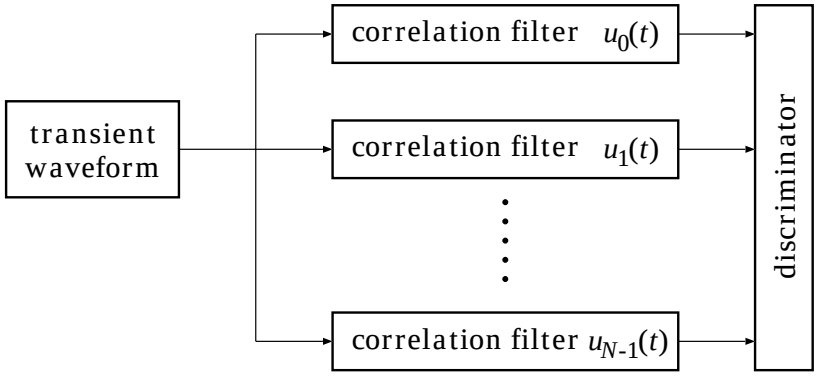


Figure 7.14: The correlation filter bank.

to become equal to zero. The other threshold $K_2(\geq 0)$ avoids a misjudgement caused by small fluctuations. The conditions $|r_i(n)| < |r_i(n+1)|$ and $|r_i(n+1)| < |r_i(n+2)|$ find out the time point when $|r_i(n)|$ reaches its maxima. The array *Time*[] holds the extracted time points of appearances of $u_i(t)$ in the transient waveform.

7.5.2 Numerical Simulations

Figure 7.15 shows the transient waveform and the outputs of the correlation filters, where the fault phase is $\phi = \pi$ rad in Eq. (7.4). The parameters are $d = 30\,\text{km}$, $T_S = 2^{-13}\,\text{ms} = 0.122\,\mu\text{s}$, $N = 4$, $K_1 = 20$, $K_2 = 0$, $L = 256$. One can see that when $u_j(t)$ appears in $v_{Sb}(t)$ the corresponding filter output $r_j(n)$ reaches its maxima as shown by a down arrow, and the other outputs are nearly equal to zero as shown by up arrows. The algorithm described above successfully extracted $u_j(t)$ for all $j = 0, 1, 2, 3$ as

$$\begin{array}{ll} \mathit{Time}[0] = \ 820T_S & \\ \mathit{Time}[1] = 2458T_S & > 1638T_S = 0.2000\,\text{ms} \\ \mathit{Time}[2] = 4096T_S & > 1638T_S = 0.2000\,\text{ms} \\ \mathit{Time}[3] = 5735T_S & > 1639T_S = 0.2001\,\text{ms} \end{array} .$$

The time intervals $1638T_S$ and $1639T_S$ correspond to fault distance 29.99 km and 30.01 km, respectively. Thus the fault point was located successfully.

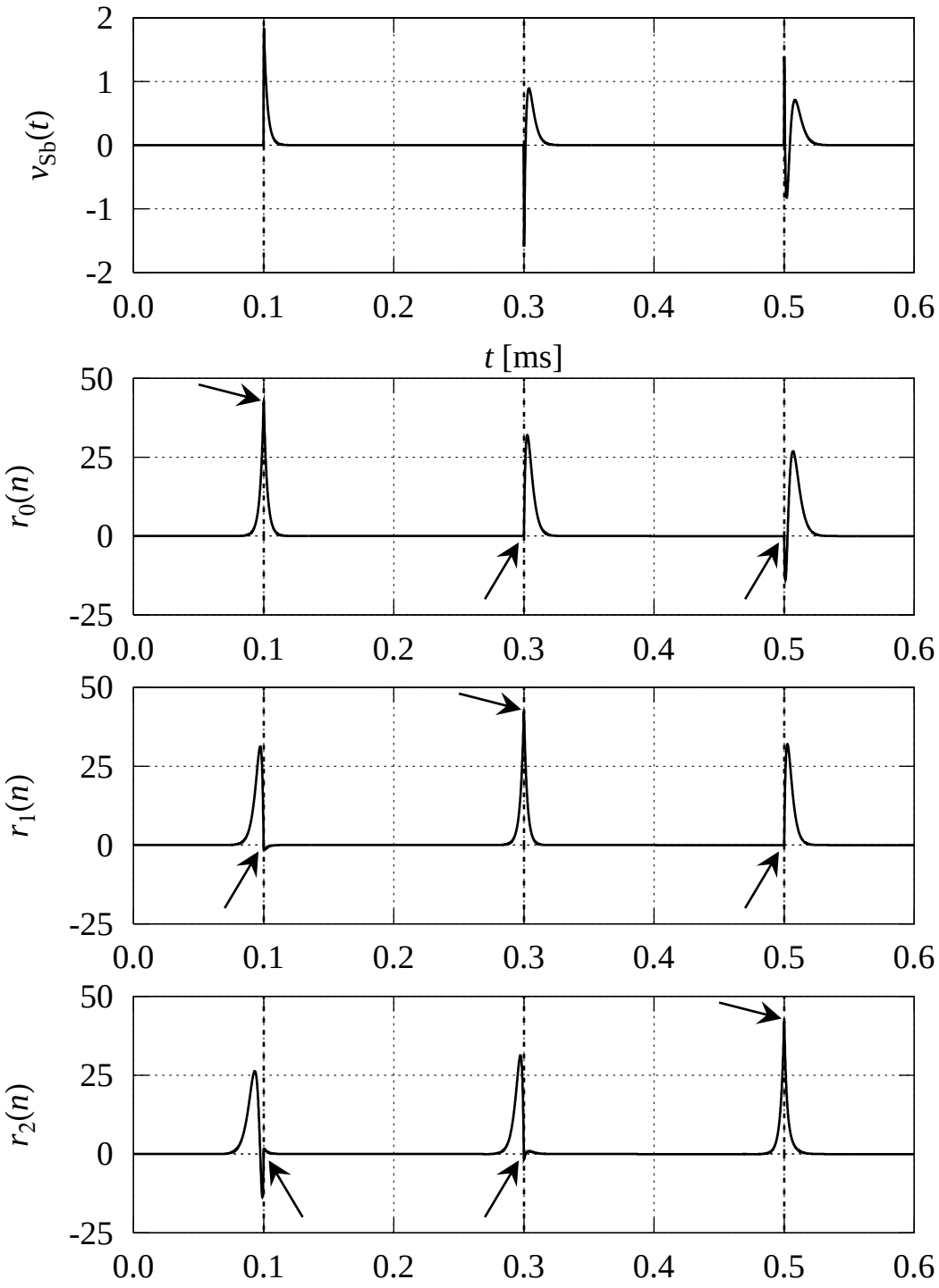


Figure 7.15: The outputs of correlation filters. The dashed vertical lines represent the extracted time points. ($T_S = 0.122\,\mu$s, $K_1 = 20, K_2 = 0, L = 256$)

7.6 Effects of a Lossy Ground

In this section, we evaluate effects of a lossy ground on the transient waveforms and the proposed fault location method by using Eqs. (6.25) and (6.26). The ground loss affects especially the earth return mode.

7.6.1 Single-Phase Transmission Lines

Figure 7.16 shows transient voltages at a fault occurrence on a single-phase transmission line above a dry ground $\sigma_g = 10^{-4}\,\mathrm{S/m}$, $\varepsilon_g = 4\varepsilon_0$, and a wet ground $\sigma_g = 10^{-3}\,\mathrm{S/m}$, $\varepsilon_g = 10\varepsilon_0$. On single-phase transmission lines, there exists only one propagation mode: the earth return mode. Hence the high frequency components of the transient waveforms are significantly attenuated by the ground losses, and as a result, the impulsive waveforms are no longer expressed by the Laguerre functions. Moreover, the pulse separation is made longer by the losses as shown in Fig. 7.16, and thereby accurate fault location is essentially difficult even if the impulsive waveforms could be extracted. Therefore, the proposed fault location method is hardly applicable to single-phase transmission lines above a lossy ground.

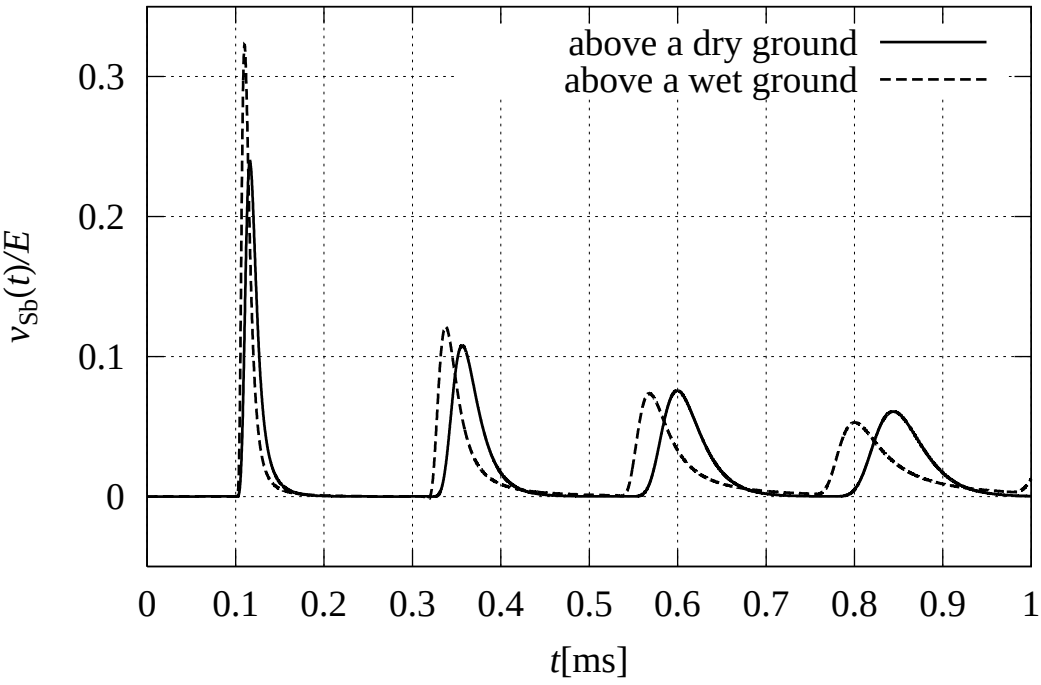


Figure 7.16: The transient waveform above a dry ground $\sigma_g = 10^{-4}\,\mathrm{S/m}, \varepsilon_g = 4\varepsilon_0$, and a wet ground $\sigma_g = 10^{-3}\,\mathrm{S/m}, \varepsilon_g = 10\varepsilon_0$.

7.6.2 Three-Phase Transmission Lines

Unlike the single-phase transmission lines, there are two aerial modes in addition to the earth return mode on three-phase transmission lines. The aerial modes concentrate around conductors and are less affected by the ground losses. The fact may make it possible to locate a fault by the proposed method.

Figures 7.17 and 7.18 show transient voltages when a single-phase-to-ground fault occurs at $d = 30\,\mathrm{km}, t = 0, \phi_1 = \pi$ between the conductor #1 and a dry ground, and a wet ground, respectively. Comparing the first impulses appearing at $t = 0.1\,\mathrm{ms}$ with those shown in Fig. 7.7, the polarity of them are altered for $v_2(0,t)$ and $v_3(0,t)$, and all of the impulses are followed by a small hump. Both of the observations show that the earth return mode propagates slower than the aerial modes and it is the common mode of the line while the aerial modes are differential modes. Though both of the earth return mode and the aerial modes are induced at the fault occurrence, the aerial modes reach the sending end earlier. Hence, the impulses are composed of the aerial modes, i.e. the differential modes, and thereby the polarity of those in $v_2(0,t)$ and $v_3(0,t)$ become negative. On the other hand, the humps are of the same polarity and originating from the earth return mode arriving at the sending end soon after.

Tables 7.1 and 7.2 present the location results for the faults above the dry/wet ground with two correlation filters, $N = 2$, and moderate thresholds, $K_1 = 3, K_2 = 2$. Figures 7.19 and 7.20 show the outputs of the correlation filters for $v_1(0,t)$. The second impulsive waveform appearing at $t = 0.23\,\mathrm{ms}$ stems from the arrival of the surge at the sending end which went toward the receiving end after the fault occurrence. The high frequency components still remain by virtue of the aerial modes so that the orthogonality of the impulsive waveforms is approximately satisfied. Thus the second impulsive waveform is successfully discriminated out by the algorithm.

phase	*Times*[0], [1]	correlation
$v_1(0,t)$	$820T_S = 0.100\,\mathrm{ms}$	$r_0(820) = 19.54$
	$2461T_S = 0.300\,\mathrm{ms}$	$r_1(2461) = 6.84$
$v_2(0,t)$	$820T_S = 0.100\,\mathrm{ms}$	$r_0(820) = -11.53$
	$2459T_S = 0.300\,\mathrm{ms}$	$r_1(2459) = -3.56$
$v_3(0,t)$	$820T_S = 0.100\,\mathrm{ms}$	$r_0(820) = -11.68$
	$2460T_S = 0.300\,\mathrm{ms}$	$r_1(2460) = -3.63$

Table 7.1: The location results for a single-phase-to-ground fault above a dry ground.

phase	*Times*[0], [1]	correlation
$v_1(0,t)$	$822T_S = 0.100\,\mathrm{ms}$	$r_0(822) = 20.32$
	$2461T_S = 0.300\,\mathrm{ms}$	$r_1(2461) = 7.11$
$v_2(0,t)$	$820T_S = 0.100\,\mathrm{ms}$	$r_0(820) = -11.85$
	$854T_S = 0.104\,\mathrm{ms}$	$r_1(854) = 3.18$
$v_3(0,t)$	$820T_S = 0.100\,\mathrm{ms}$	$r_0(820) = -11.81$
	$853T_S = 0.104\,\mathrm{ms}$	$r_1(853) = 3.23$

Table 7.2: The location results for a single-phase-to-ground fault above a wet ground.

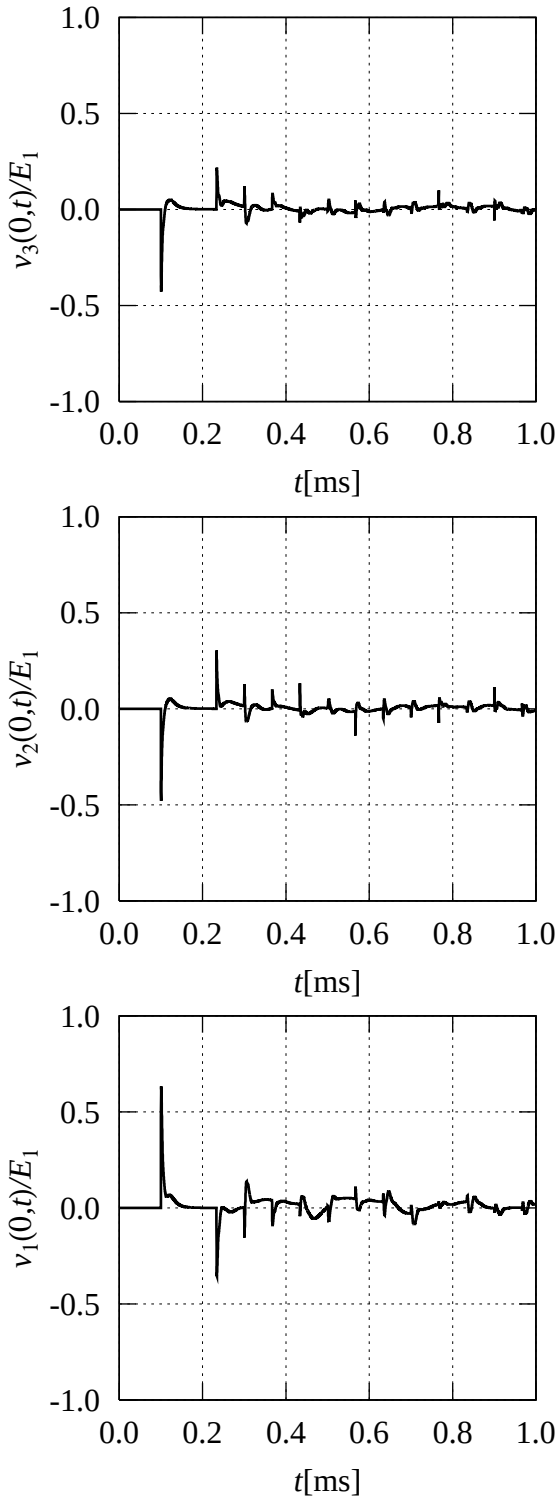


Figure 7.17: Transient waveforms at a single-phase-to-ground fault above a dry ground.

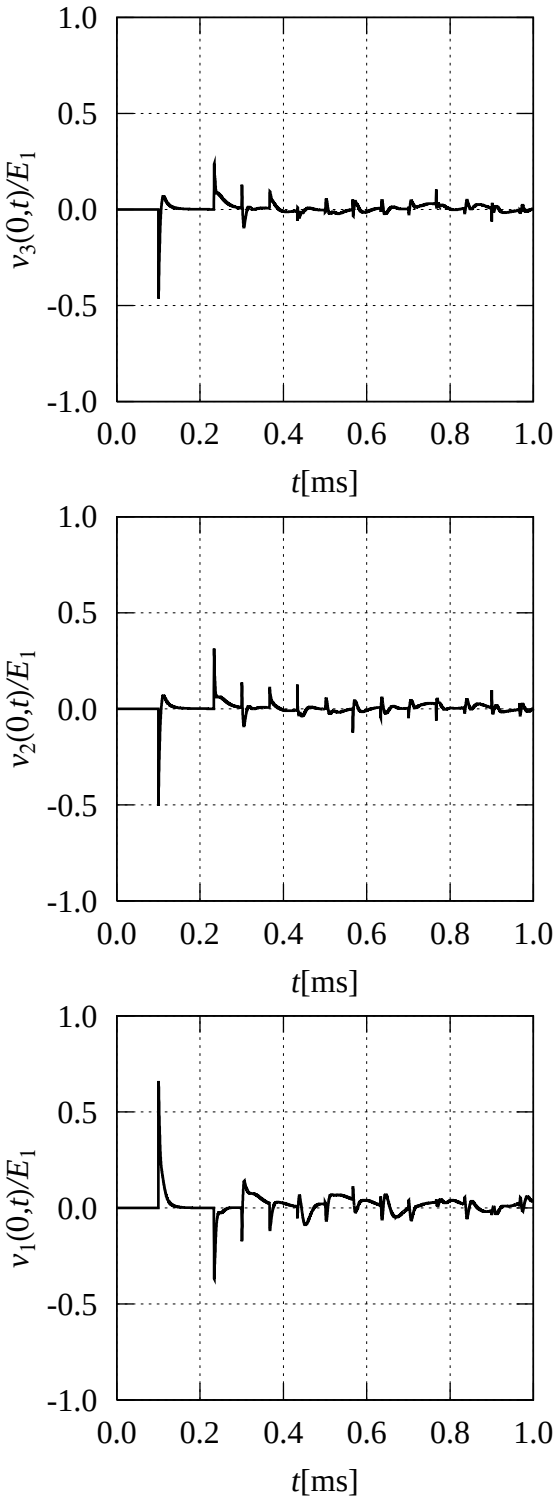


Figure 7.18: Transient waveforms at a single-phase-to-ground fault above a wet ground.

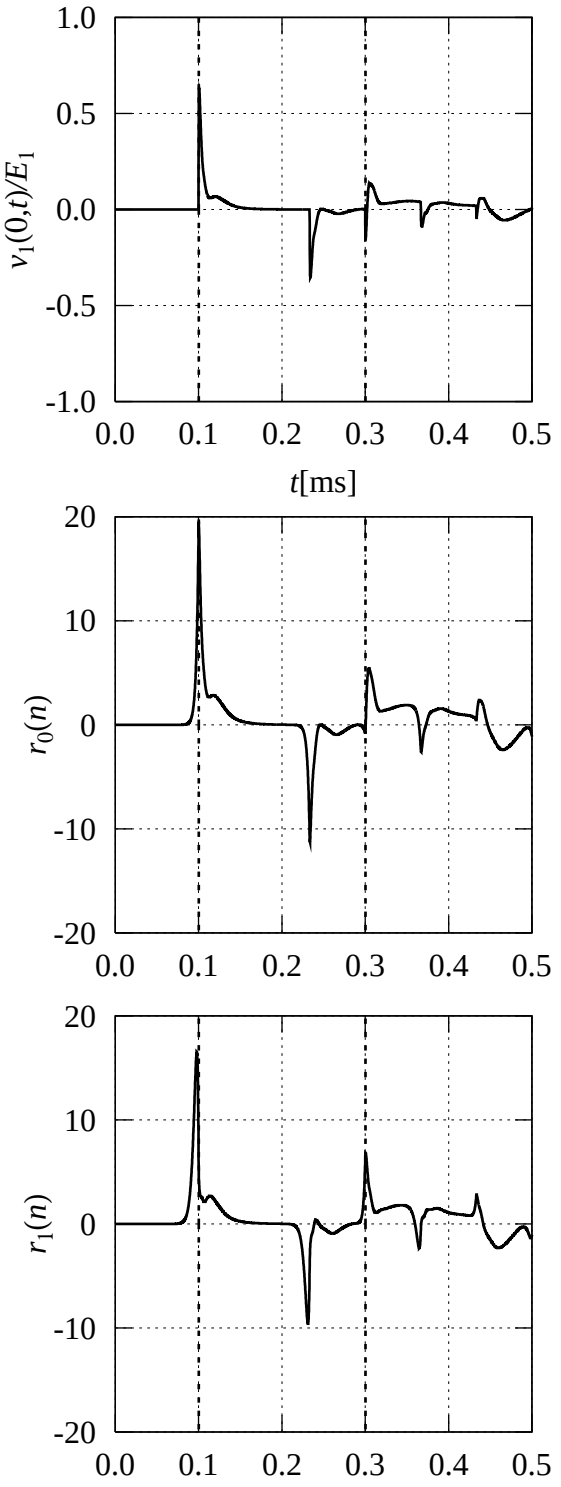


Figure 7.19: The outputs of correlation filters at a single-phase-to-ground fault above a dry ground. ($T_S = 0.122\,\mu$s, $N = 2, K_1 = 3, K_2 = 2, L = 256$)

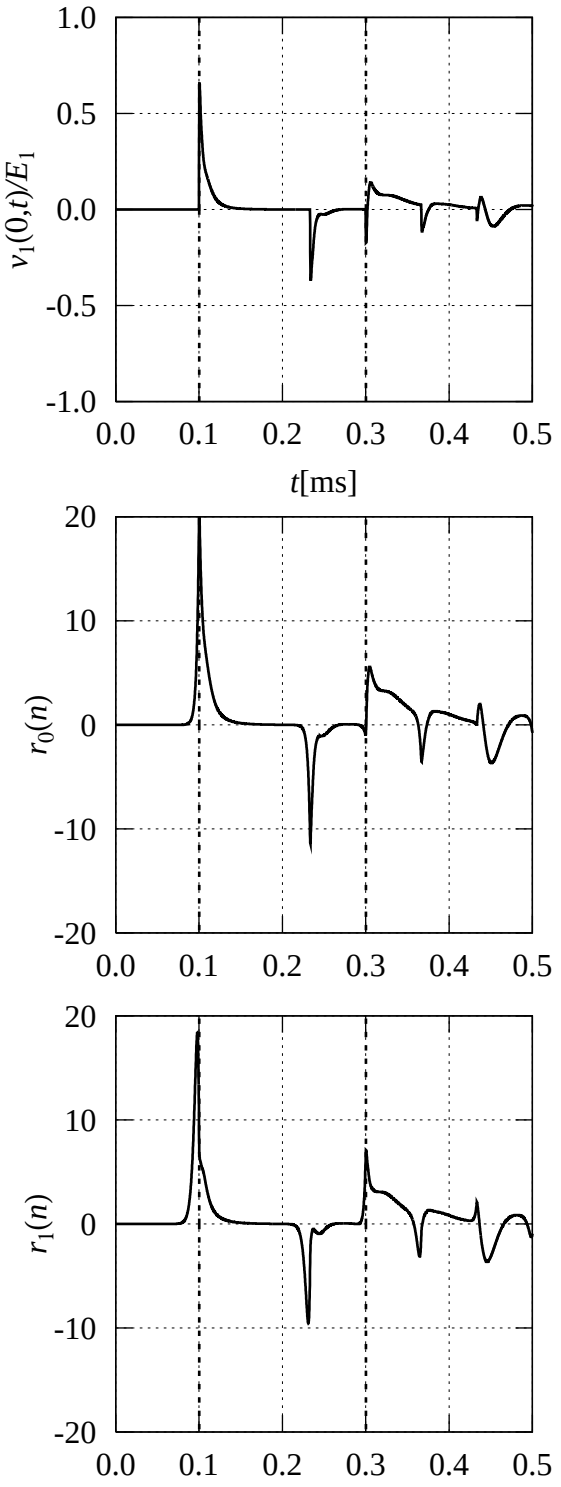


Figure 7.20: The outputs of correlation filters at a single-phase-to-ground fault above a wet ground. ($T_S = 0.122\,\mu$s, $N = 2, K_1 = 3, K_2 = 2, L = 256$)

The $u_1(t)$ was not extracted properly for $v_2(0,t), v_3(0,t)$ as shown in Table 7.2. By using the results for $v_1(0,t)$, the fault point is located as

$$\frac{(Time[1] - Time[0])c}{2} = \frac{1641 T_S c}{2} = 30.05\,\text{km}, \tag{7.34}$$

$$\frac{(Time[1] - Time[0])c}{2} = \frac{1639 T_S c}{2} = 30.01\,\text{km}, \tag{7.35}$$

for the dry ground and the wet ground, respectively.

Similarly, Figs. 7.21 and 7.22 show transient voltages when a phase-to-phase fault occurs at $d = 30\,\text{km}, t = 0, \phi_2 = \pi$ between the conductors #1 and #2 above a dry ground and above a wet ground, respectively. In the case of a phase-to-phase fault with $\phi_2 = \pi$, almost no earth return mode is excited since $v_1(d,0) + v_2(d,0) + v_3(d,0) = 0$, and then the transient waveforms are much less affected by the ground losses.

Tables 7.3 and 7.4 present the location results for the faults above the dry/wet ground with two correlation filters, $N = 2$, and moderate thresholds, $K_1 = 3, K_2 = 2$. Although there are small impulsive waveforms seen in $v_3(0,t)$, they are discriminated out by the threshold K_2. Figures 7.23 and 7.24 show the outputs of the correlation filters for the fault phase, $v_1(0,t)$. By using the results for $v_1(0,t)$, the fault point is located as

$$\frac{(Time[1] - Time[0])c}{2} = \frac{1640 T_S c}{2} = 30.03\,\text{km}, \tag{7.36}$$

$$\frac{(Time[2] - Time[1])c}{2} = \frac{1639 T_S c}{2} = 30.01\,\text{km}. \tag{7.37}$$

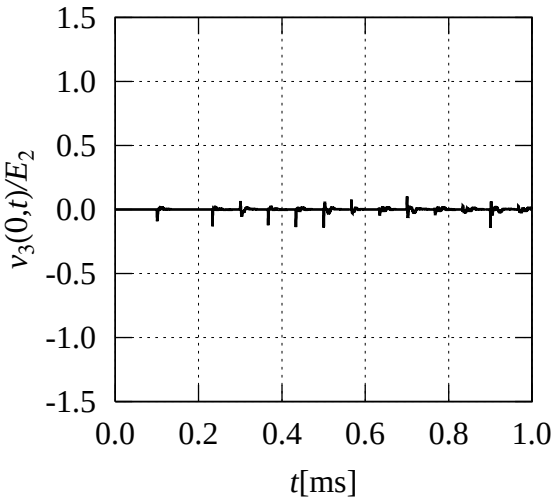
for the dry ground and the wet ground respectively.

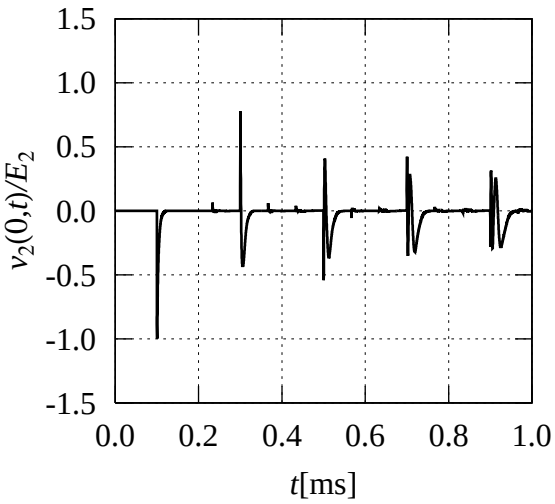
phase	*Times*[0], [1]	correlation
$v_1(0,t)$	$820T_S = 0.100\,\text{ms}$	$r_0(820) = 23.61$
	$2460T_S = 0.300\,\text{ms}$	$r_1(2460) = 21.19$
$v_2(0,t)$	$820T_S = 0.100\,\text{ms}$	$r_0(820) = -25.87$
	$2459T_S = 0.300\,\text{ms}$	$r_1(2459) = -23.50$
$v_3(0,t)$	—	—
	—	—

Table 7.3: The location results for a phase-to-phase fault above a dry ground.

phase	*Times*[0], [1]	correlation
$v_1(0,t)$	$820T_S = 0.100\,\text{ms}$	$r_0(820) = 23.44$
	$2459T_S = 0.300\,\text{ms}$	$r_1(2459) = 21.11$
$v_2(0,t)$	$820T_S = 0.100\,\text{ms}$	$r_0(820) = -26.13$
	$2459T_S = 0.300\,\text{ms}$	$r_1(2459) = -23.93$
$v_3(0,t)$	—	—
	—	—

Table 7.4: The location results for a phase-to-phase fault above a wet ground.





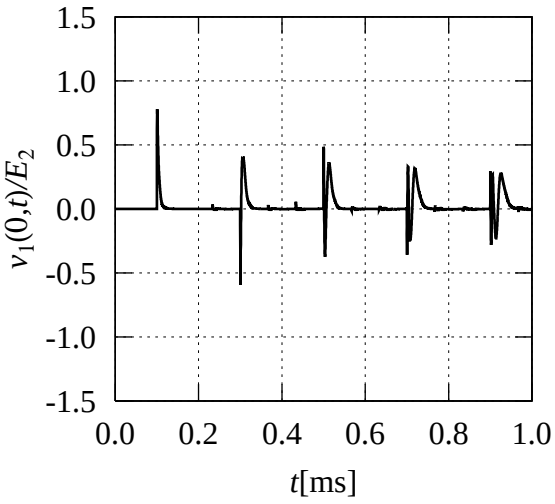
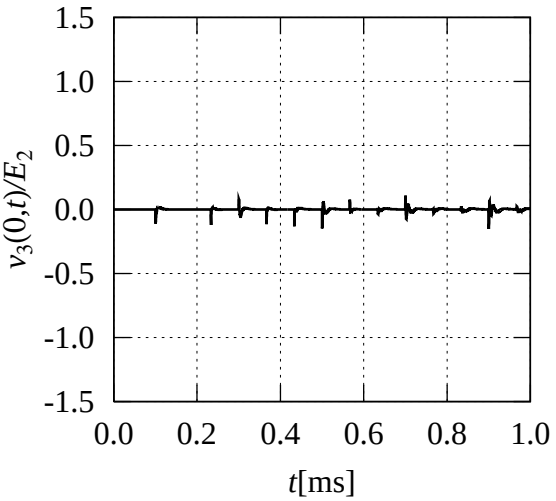
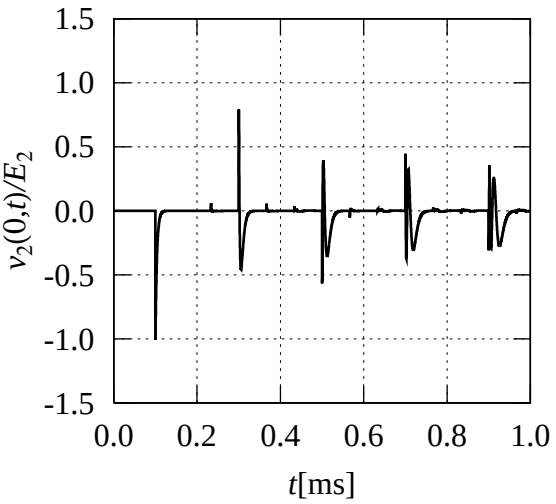


Figure 7.21: Transient waveforms at a phase-to-phase fault above a dry ground.





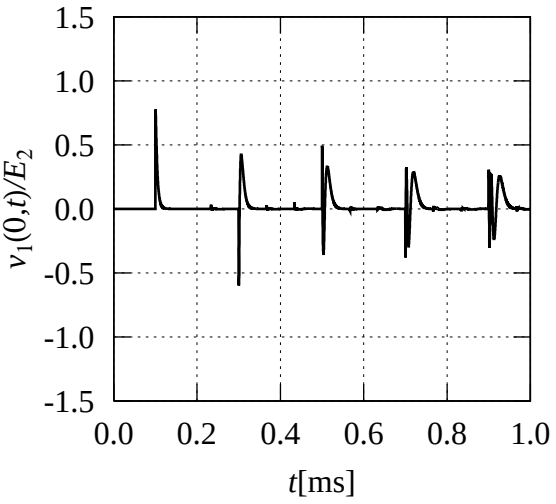


Figure 7.22: Transient waveforms at a phase-to-phase fault above a wet ground.

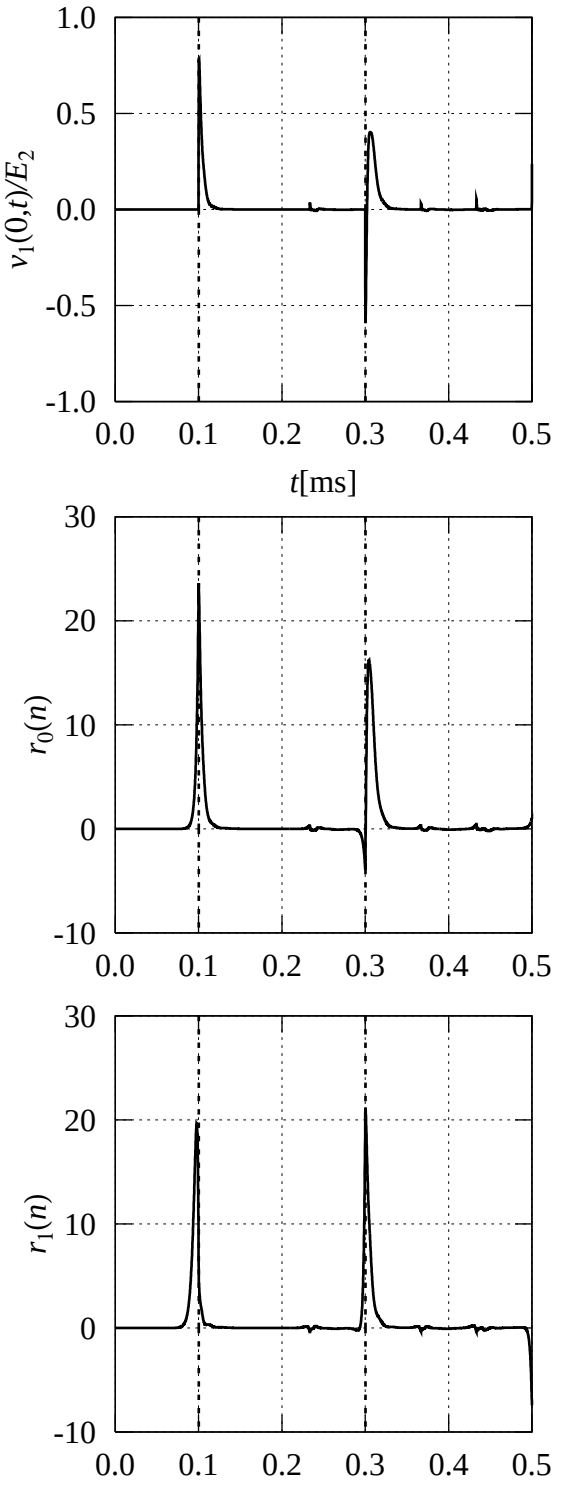


Figure 7.23: The outputs of correlation filters at a phase-to-phase fault above a dry ground. ($T_S = 0.122\,\mu s, N = 2, K_1 = 3, K_2 = 2, L = 256$)

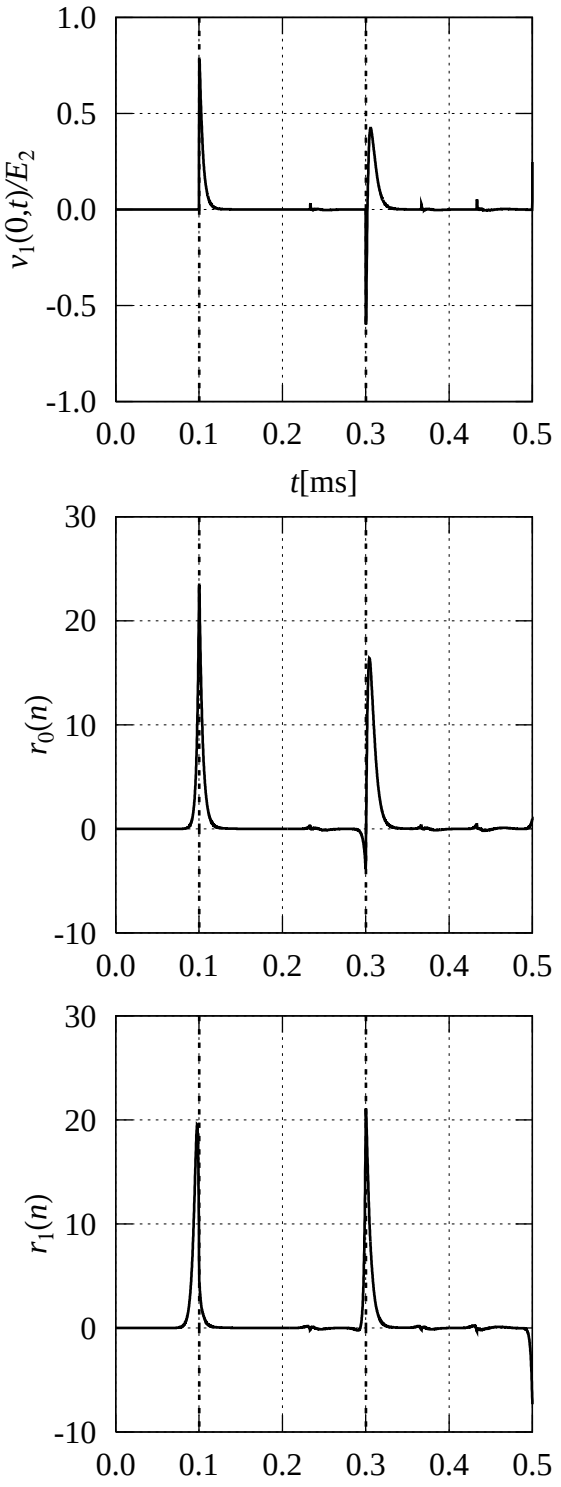


Figure 7.24: The outputs of correlation filters at a phase-to-phase fault above a wet ground. ($T_S = 0.122\,\mu s, N = 2, K_1 = 3, K_2 = 2, L = 256$)

7.7 Concluding Remarks

We showed with some simplifications that the impulsive waveforms observed at a fault occurrence are expressed by the well-known Laguerre functions. The case of a single-phase transmission line is fundamental, in which j times trips of the surge between the sending end and the fault point originate the j-th Laguerre function observed at the sending end as the j-th impulsive waveform. For an ungrounded multi-phase transmission line, though a single-phase-to-ground fault involves complicated multiple reflections, the impulsive waveforms are also found to be the Laguerre functions. The case of a phase-to-phase fault simply reduces to the case of single-phase transmission lines.

Making use of the orthogonality of the Laguerre functions, we demonstrated the possibility of a novel method of fault location by using a correlation filter bank. The filter bank discriminates the impulsive waveforms by computing correlations between the transient waveform and precomputed impulsive waveforms. The method successfully located a fault that occurred on a single-phase transmission line above a perfectly conducting ground.

Furthermore, we investigated effects of a lossy ground which affects especially the earth return mode. For single-phase transmission lines, there exists only the earth return mode and thereby the impulsive waveforms are significantly distorted and are no longer expressed by the Laguerre functions. On the other hand, for three-phase transmission lines, two aerial modes enable propagation of high frequency components and the orthogonality of the impulsive waveforms are barely retained for first several waveforms. Numerical simulations suggest that the fault location by the proposed method is still possible with moderate thresholds.

We considered the short circuits thus far. What if the line is cut off at $x = d$? From the viewpoint of circuit theory, a cutoff is the dual of a short. Then, by considering a current source $\tilde{i}(t)$ instead of the voltage source $\tilde{e}(t)$ at the fault point, we still have the impulsive voltage waveforms appearing at the sending end. Note that the current source is the dual of the voltage source.

Chapter 8

Conclusion

In this thesis, we analytically investigated the FFT-based numerical Laplace transforms, and proposed more accurate and efficient methods. The proposed method was applied to the transient analysis of transmission lines at fault occurrences. We found out that the transient waveforms are expressed by the Laguerre functions, and proposed a novel method of fault location which utilizes the orthogonality of the Laguerre functions.

In Chapter 2, we introduced the conventional FFT-based numerical Laplace transforms. Then, in Chapter 3, we improved the FFT-based Laplace inversion. We first analyzed errors incurred by the FFT-based Laplace inversion, and showed that the large errors are mostly composed of the truncation error, whose amount depends on the asymptotic behavior of the Laplace transform to be inverted. Then, we proposed a simple yet effective acceleration method: the Laplace transform is first multiplied with $1/s^i$ to suppress its high-frequency components, and after the FFT-based inversion, the obtained inversion is i-th differentiated in the time domain. Moreover, we analyzed errors introduced by the proposed method especially for the unit step function, and showed $i = 3 \sim 4$ is the most suitable due to the round off errors of the numerical differences. Numerical examples showed that the proposed method remarkably reduces the truncation errors, avoids Gibbs phenomena by removing discontinuities through the integration, and extends valid region to almost the entire region.

In Chapter 4, we proposed a novel numerical Laplace transform pair. Based on the analysis of the conventional numerical Laplace transform pair, we showed that interpolations used in the conventional pair correspond to extrapolations in the other domain, and interpolated data points are soon discarded in the subsequent step. By eliminating the interpolations, we constructed the no-interpolation pair between $f_{\mathrm{P}}(n\Delta t)$ whose latter half is filled with zeros and $F_{\mathrm{P}}(a + \mathrm{j}k\Delta\omega)$

whose latter half is filled with the reversed complex conjugate of the former half. Then, the no-interpolation pair turned out to make a numerically reversible pair if $f_P(t)$ is given first and $f_P(0) = 0$. Furthermore, we proposed a novel transform pair by employing the inversion proposed in Chapter 3 in the no-interpolation pair. The proposed pair extended the valid region nearly twice, and gave more accurate and computationally efficient transform pair.

In Chapter 5, we designed a processor which performs numerical forward/inverse Laplace transformations on an FPGA. We first designed the processor with fixed-point numbers and encountered the severe lack of precision especially in the case of inversions. The lack of precision stemmed from the combination of the poor magnitude content of $F(s)$ and the use of fixed-point numbers in the FFT computation. Moreover, the wide dynamic range of the exponential function limited the precision of the inversion. Then, instead of using the floating-point numbers, we introduced the block-floating-point numbers into the FFT computation, and eliminated unneeded divisions by two that were used to prevent an overflow. As a result, the precision was improved for inverse transformations by several bits nearly to the inherent limit posed by the exponential function, and for forward transformations by a few bits nearly to the full 16-bit precision.

In Chapter 6, we introduced analyzing method of transmission lines in the Laplace domain. Then, in Chapter 7, we analyzed transients on transmission lines at a fault occurrence. With some simplifications, we analytically showed that the impulsive waveforms, which are observed at the sending end after a fault occurrence on a single-phase transmission line, are identical to the well-known Laguerre functions. The same argument was derived for a single-phase-to-ground fault and a phase-to-phase fault on an ungrounded multi-phase transmission line. Then, we proposed a novel method of fault location by making use of the orthogonality of the Laguerre functions. The proposed method successfully located a fault that occurred on a single-phase transmission line above a perfectly conducting ground. Moreover, we numerically evaluated the effects of a lossy ground on the fault location by using the proposed numerical inversion of Laplace transforms. In the case of single-phase transmission lines, the impulsive waveforms were significantly affected by the ground losses and the fault location was hardly possible. However, in the case of three-phase transmission lines, by virtue of the existence of the two aerial modes, high frequency components could propagate without being attenuated, and consequently the fault location was still possible.

Acknowledgments

I wish to express my sincere gratitude to Professor Kohshi Okumura at Kyoto university for his continuous guidance since I joined the Okumura laboratory in the department of electrical engineering when I was a senior student, and for his encouragement throughout this study.

I would like to thank Associate Professor Takashi Hisakado at Kyoto university, who has been leading interesting seminars and reading groups, for his in-depth discussions and invaluable advices.

I am also thankful to Doctor Satoshi Ichikawa for his precious comments.

Special thanks are due to the members of the laboratory for their support.

Bibliography

[1] B. Davies and B. Martin, "Numerical inversion of the Laplace transform: a survey and comparison of methods," *J. Comp. Phys.*, vol. 33, pp. 1–32, 1979.

[2] B. Davies, *Integral Transforms and Their Applications*. Springer Verlag New York, third ed., 2002.

[3] R. E. Bellman, R. E. Kalaba, and J. Lockett, *Numerical Inversion of the Laplace Transform*. New York: American Elsevier Publishing Company, 1966.

[4] W. T. Weeks, "Numerical Inversion of Laplace Transforms using Laguerre Functions," *J. ACM*, vol. 13, no. 3, pp. 419–426, 1966.

[5] R. Piessens, "Gaussian quadrature formulas for the numerical integration of Bromwich's integral and the inversion of the Laplace transform," *J. Engirg. Math.*, vol. 5, no. 1, pp. 1–9, 1971.

[6] A. Talbot, "The accurate numerical inversion of Laplace transforms," *J. Inst. Math. Appl.*, vol. 23, pp. 97–120, 1979.

[7] H. Arita and A. Ametani, "Study of exponential Laplace transform and its application to analyses of transient phenomena," *IEEJ Trans. C*, vol. 102-C, pp. 77–84, 1982. (Japanese).

[8] H. Dubner and J. Abate, "Numerical inversion of Laplace transforms by relating them to the finite Fourier Cosine transform," *J. ACM*, vol. 15, no. 1, pp. 115–123, 1968.

[9] F. Durbin, "Numerical inversion of Laplace transforms: an efficient improvement to Dubner and Abate's method," *Computer J.*, vol. 17, no. 4, pp. 371–376, 1974.

[10] S. Ichikawa and A. Kishima, "Applications of Fourier series technique to inverse Laplace transform II," *Mem. Fac. Eng. Kyoto Univ.*, vol. 35, pp. 393–400, 1973. (Japanese).

[11] F. Veillon, "Numerical inversion of Laplace transform," *Commun. ACM*, vol. 17, no. 10, pp. 587–591, 1974.

[12] K. S. Crump, "Numerical inversion of Laplace transforms using a Fourier series approximation," *J. ACM*, vol. 23, no. 1, pp. 89–96, 1976.

[13] J. T. Hsu and J. S. Dranoff, "Numerical inversion of certain Laplace transforms by the direct application of fast Fourier transform (FFT) algorithm," *Comput. & Chem. Eng.*, vol. 11, no. 2, pp. 101–110, 1987.

[14] T. Hosono, "Numerical inversion of Laplace transform," *IEEJ Trans. A*, vol. 99, no. 10, pp. 494–500, 1979. (Japanese).

[15] D. J. Wilcox, "Numerical Laplace transformation and inversion," *Int. J. Elect. Enging Educ.*, vol. 15, pp. 247–265, 1978.

[16] D. J. Wilcox and I. S. Gibson, "Numerical Laplace transformation and inversion in the analysis of physical systems," *Int. J. Numer. Meth. Eng.*, vol. 20, pp. 1507–1519, 1984.

[17] D. J. Wilcox and I. S. Gibson, "Evaluation of a new method for numerical Laplace transformation and inversion," *Int. J. Numer. Meth. Eng.*, vol. 20, pp. 1521–1528, 1984.

[18] K. Okumura, A. Kishima, and S. Tokoro, "A method for computing electrical transients of transmission lines by numerical Laplace transform," *IECE Trans. A*, vol. J68-A, no. 2, pp. 107–114, 1985. (Japanese. English translation available).

[19] A. V. Oppenheim and C. J. Weinstein, "Effects of finite register length in digital filtering and the fast Fourier transform," *Proc. IEEE*, vol. 60, pp. 957–975, 1972.

[20] C. R. Paul, *Analysis of multiconductor transmission lines*. New York: Wiley Interscience, 1994.

[21] S. Hayashi, *Surges on Transmission Systems*. Denki-Shoin, 1955.

[22] Gantmacher, *Matrix Theory Vol.I*. Chelsea, 1959.

[23] J. R. Carson, "Wave propagation in overhead wires with ground return," *Bell System Technical Journal*, vol. 5, pp. 539–554, 1926.

[24] E. D. Sunde, *Earth conduction effects in transmission systems*. Von Nostrand, New York, 1949.

[25] H. Kikuchi, "Electromagnetic fields on infinite wire above plane-earth at high frequencies," *Bulletin of the Electrotechnical Laboratory*, vol. 21, no. 6, pp. 439–454, 1957. Japanese.

[26] S. Ichikawa, "Numerical analysis of traveling waves in transmission networks by numerical inversion of Laplace transform," *IEEJ Trans. B*, vol. 102-B, no. 12, pp. 785–792, 1982. Japanese.

[27] C. Moler and C. van Loan, "Nineteen dubious ways to compute the exponential of a matrix," *SIAM Review*, vol. 20, no. 4, pp. 801–836, 1978.

[28] Y. W. Lee, "Synthesis of electrical networks by means of the Fourier transforms of Laguerre's functions," *J. Math. Phy. MIT*, vol. 11, no. 2, pp. 83–113, 1932.

[29] Z. Zang, "Iterative algorithms for envelope constrained recursive filter design via Laguerre functions," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 46, no. 11, pp. 1342–1348, 1999.

[30] R. E. King and P. N. Paraskevopoulos, "Digital Laguerre Filters," *Int. J. Circuit Theory and Applications*, vol. 5, pp. 81–91, 1977.

[31] B. Wahlberg, "System identification using Laguerre models," *IEEE Transactions on Automatic Control*, vol. 36, pp. 551–562, 1991.

[32] H. W. Dommel and J. M. Michels, "High speed relaying using traveling wave transient analysis," in *IEEE Publications No.78CH1295-5 PWR*, no. A78 214-9, (New York), pp. 1–7, IEEE PES Winter Power Meeting, January 1978.

[33] M. Chamia and S. Liberman, "Ultra high speed relay for ehv/uhv transmission lines — development, design, and application," *IEEE Trans. PAS*, vol. 97, no. 6, pp. 2104–2116, 1978.

[34] G. W. Swift, "The spectra of fault-induced transients," *IEEE Trans. PAS*, vol. 98, no. 3, pp. 940–947, 1979.

[35] A. T. Johns, “New ultra-high-speed directional comparison technique for the protection of e.h.v. transmission lines,” *IEE Proc.C*, vol. 127, no. 4, pp. 228–239, 1980.

[36] M. Vitins, “A fundamental concept for high speed relaying,” *IEEE Trans. PAS*, vol. 100, no. 1, pp. 163–173, 1981.

[37] P. A. Crossley and P. G. McLaren, “Distance protection based on travelling waves,” *IEEE Trans. PAS*, vol. 102, no. 9, pp. 2971–2983, 1983.

[38] S. Rajendra and P. G. McLaren, “Travelling-wave techniques applied to the protection of teed circuits: principle of travelling wave techniques,” *IEEE Trans. PAS*, vol. 104, no. 12, pp. 3544–3550, 1985.

[39] E. H. Shehab-Eldin and P. G. McLaren, “Travelling wave distance protection – problem areas and solutions,” *IEEE Trans. on Power Delivery*, vol. 3, no. 3, pp. 894–902, 1988.

[40] A. A. Fernando, H. Magnago, “Fault location using wavelets,” *IEEE Trans. on Power Delivery*, vol. 13, no. 4, pp. 1475–1480, 1998.

[41] E. Styvaktakis, M. H. J. Bollen, and I. Y. H. Gu, “A fault location technique using high frequency fault clearing transients,” *IEEE Power Eng. Rev*, vol. 19, no. 5, pp. 58–60, 1999.

Publications

Journal Publications

- A. Yonemoto, T. Hisakado, and K. Okumura, "Expression of transient phenomena at faults on ideal transmission lines by Laguerre functions," *IEE Proc. Circuits Devices Syst.*, vol. 150, no. 2, pp. 141–147, 2003.

- A. Yonemoto, T. Hisakado, and K. Okumura, "Accuracy improvement of the FFT-based numerical inversion of Laplace transforms," *IEE Proc. Circuits Devices Syst.*, vol. 150, no. 5, pp. 399–404, 2003.

Conference Publications

- A. Yonemoto and K. Okumura, "Analytical expressions for transient waveforms at faults on transmission lines and its application," *Convention records of annual meeting of Electric-Related Institutes in Kansai district*, no. G152, 1999. Japanese.

- A. Yonemoto, T. Hisakado, and K. Okumura, "Transient analysis of transmission lines at short failure and construction of filters," *IEICE Technical Report*, no. NLP2000-82, pp. 23–30, 2000. Japanese.

- A. Yonemoto, T. Hisakado, and K. Okumura, "An improvement of convergence of FFT-based numerical inversion of Laplace transform," *Record of the Kansai-section Joint Convention of Institutes of Electrical Engineering*, no. G3, 2001. Japanese.

- A. Yonemoto, T. Hisakado, and K. Okumura, "Representation of transient waveforms at faults on three-phase transmission lines by Laguerre polynomials," *Proc. IEICE General Conference*, no. A-1-59, 2001. Japanese.

- A. Yonemoto, T. Hisakado, and K. Okumura, “An improvement of convergence of FFT-based numerical Laplace transformation,” *IEICE Technical Report*, no. DSP2001-188, 2002.
- A. Yonemoto, T. Hisakado, and K. Okumura, “A processor for inversion of Laplace transforms with pseudo-floating-point numbers,” *Proc. IEICE General Conference*, no. A-4-57, 2002. Japanese.
- A. Yonemoto, T. Hisakado, and K. Okumura, “Fault location of single-phase transmission lines by Laguerre function,” *IEEE Proc. ISCAS 2002*, vol. I, pp. 161–164, 2002.
- A. Yonemoto, T. Hisakado, and K. Okumura, “An improvement of convergence of FFT-based numerical inversion of Laplace transforms,” *IEEE Proc. ISCAS 2002*, vol. V, pp. 769–772, 2002.
- A. Yonemoto, T. Hisakado, and K. Okumura, “An implementation of numerical inversion of Laplace transforms on FPGA,” *IEEE Proc. ISCAS 2003*, vol. IV, pp. 492–495, 2003.
- A. Yonemoto, T. Hisakado, and K. Okumura, “Improvements on efficiency and accuracy of numerical Laplace transform pair,” *Proc. IEICE Society Conference*, no. SA-2-6, 2003. Japanese.